University of Alberta

SIMPLE KINEMATICS-BASED ROBOT IMITATION LEARNING OF REPETITIVE MOTOR TASKS THROUGH VISUAL PERCEPTION

by

Axel von Bertoldi

©

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Master of Science**.

Department of Electrical and Computer Engineering

Edmonton, Alberta
Fall 2003

National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisisitons et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-612-87904-6
Our file Notre référence
ISBN: 0-612-87904-6

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou aturement reproduits sans son autorisation.

Canada

University of Alberta

Library Release Form

Name of Author: Axel von Bertoldi

Title of Thesis: Simple Kinematics-based Robot Imitation Learning of Repetitive Motor Tasks Through Visual Perception

Degree: Master of Science

Year this Degree Granted: 2003

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as herein before provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.

Date: 27/08/03

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Simple Kinematics-based Robot Imitation Learning of Repetitive Motor Tasks Through Visual Perception** submitted by Axel von Bertoldi in partial fulfillment of the requirements for the degree of **Master of Science**

Date: August 21, 2003

Abstract

This thesis presents an approach to robot learning based on visually mediated imitation. Imitation learning enables robots to acquire new behaviors simply by observation of other agents performing the behavior, and can also improve performance of other robot learning approaches. Imitation learning has already allowed humanoid robots to acquire a number of discrete skills. This work extends previous research by considering rhythmic skills, and by attempting to quantify the accuracy with which the new task is acquired. Visual mediation is accomplished in two steps: 1.) use of a highly accurate optical motion capture system to perceive and record motion, and 2.) subsequent application of methods borrowed from computer animation and bio-mechanical analysis to extract kinematic information from the motion capture data. Rhythmic motion is represented by means of a Matsuoka neural oscillator. A simple learning strategy is implemented as a least-squares minimization of the error between the observed and recreated joint angles. The proposed approach is evaluated by applying it to a drumming task, using mean squared error and frequency analysis on the observed and recreated end effector trajectories as metrics of accuracy.

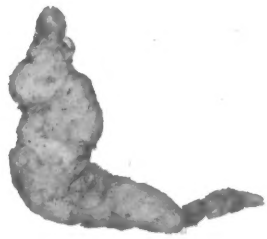
"A witty saying proves nothing."

- Voltaire (1694-1778)

To my parents, for giving me the opportunity to pursue an education by coming to Canada

Acknowledgements

Thanks are in order to Adrian Liggins, Justin Lewicke, Jim Raso, and Terry Blois for allowing me unsupervised access to the new motion capture system at the Glenrose Hospital. Thanks to my parents and sister for the support and encouragement, and thanks to Joss for the multifaceted support and help.



Contents

1 Introduction 1
- 1.1 Overview . . . 1
- 1.2 Approach and Direction . . . 2
- 1.3 Thesis outline . . . 3
- 1.4 Contributions . . . 3

2 Related Work 5
- 2.1 Visual Perception of Motion . . . 5
  - 2.1.1 Motion perception and Capture . . . 5
  - 2.1.2 Attention Focus . . . 9
  - 2.1.3 Generating robot kinematic data from motion capture data . . . 10
  - 2.1.4 Correspondence of kinematics . . . 11
- 2.2 Control policies for imitation learning . . . 12
- 2.3 Imitation learning modes . . . 14
- 2.4 Imitation learning testbeds . . . 16
  - 2.4.1 Humanoid robots and manipulators . . . 16
  - 2.4.2 Simulations . . . 17
- 2.5 Biological Inspiration for imitation learning . . . 17

3 Human Motion Measurement 19
- 3.1 Introduction . . . 19
- 3.2 Motion Capture Systems . . . 19
- 3.3 Experimental equipment . . . 19
  - 3.3.1 Motion Analysis Passive Optical Motion Capture System . . . 20
  - 3.3.2 System performance . . . 24
  - 3.3.3 Additional Hardware . . . 26
- 3.4 Capturing human motion . . . 28
  - 3.4.1 Marker set . . . 28
  - 3.4.2 Complete setup . . . 29
  - 3.4.3 Capture trials . . . 31
- 3.5 Results . . . 33
- 3.6 Conclusions . . . 33

4 Generating Robot Kinematic Data From Optical Motion Capture Systems 35
- 4.1 Introduction . . . 35
- 4.2 Issues in generating kinematic data from optical motioncapture system . . . 36
- 4.3 Methods . . . 41
- 4.4 Implementation . . . 51
- 4.5 Results . . . 51
  - 4.5.1 Human accuracy in repetitive motor tasks . . . 60
- 4.6 Conclusions . . . 63

5 Analysis of Human Motion 64
5.1 Introduction . . . 64
5.2 Methods . . . 64
5.3 Results . . . 67
5.4 Conclusions . . . 69

6 Rhythmic motion though Matsuoka neural oscillators 71
6.1 Introduction . . . 71
6.2 The Matsuoka neural oscillator . . . 71
6.2.1 Oscillator behavior and properties . . . 72
6.2.2 Neural oscillators as motion primitives . . . 77
6.2.3 Networks of oscillators for multi-joint control . . . 79
6.3 Methods . . . 80

7 Imitation learning 86
7.1 Introduction . . . 86
7.2 Methods . . . 86
7.2.1 Non-linear oscillators as control policies . . . 86
7.2.2 Learning of drumming task . . . 87
7.3 Implementation . . . 88
7.4 Results . . . 89
7.4.1 Sources of Error . . . 95
7.5 Conclusions . . . 96

8 Conclusions and Further Work 97
8.1 Conclusions . . . 97
8.2 Improvements to current methods . . . 98
8.3 Review of thesis . . . 101
8.4 Contributions . . . 101
8.5 Further work . . . 102

Bibliography 105

A Analysis Results of Single Stroke Drumming Task 110
A.1 Subject one results . . . 111
A.2 Subject two results . . . 115
A.3 Subject three results . . . 119

B Learning Results 123
B.1 Subject one results . . . 124
B.2 Subject two results . . . 126
B.3 Subject three results . . . 128

List of Figures

3.1 Motion Analysis Eagle digital motion capture camera. 21
3.2 Configuration of the eight-camera motion capture system at Glenrose Hospital. . . 21
3.3 Passive retro-reflective markers used with motion capture system. 22
3.4 Device used by Richards to test the accuracy of a motion capture systems. 25
3.5 Drumming force pad, shown with force sensors attached. 27
3.6 Force sensor conditioning circuit: current to voltage converter, and low pass filter. . 27
3.7 Input voltage vs force calibration curve for three force sensors. 28
3.8 Standard full body bio-mechanical marker set recommended by Motion Analysis. . 29
3.9 Subject with passive retro-reflective markers attached. 31
3.10 Partial time series of stick tip real marker trajectory. 33
3.11 Partial time series of drum impact force. 34

4.1 EVaRT virtual marker visualization and definition tool. 47
4.2 Trajectory of marker rigidly attached to segment with respect to parent segment. . . 49
4.3 Graphical user interface used to initiate all tasks. 52
4.4 Animated model of robot created using Matlab robotics toolbox. 52
4.5 Recovered joint angles for first gym motion trial for subject two. 56
4.6 Example measured (- ■ -) and recreated (-•-) motion for first gym trial. 57
4.7 Example of residual error at each joint for first gym trial of subject 2. 59
4.8 Actual length of all segments during first gym motion trial subject two. 61
4.9 Drumming impact coordinates (subject 2, 6.13 Hz, medium dynamic level). 62

5.1 Angle trajectories for single stroke drumming task at 4 Hz, subject one. 65
5.2 Partial time series of drumstick-tip marker Z component trajectory. 66

6.1 Schematic of Matsuoka mutually inhibiting neurons. 72
6.2 Matsuoka neural oscillator as dynamic system. 73
6.3 Oscillator output with no input, oscillating at its natural frequency and amplitude. . 74
6.4 Oscillator at various stages of entrainment to an input. 74
6.5 Oscillator output amplitude plotted as a function of input frequency. 76
6.6 Oscillator gain and phase as a function of input frequency. 76
6.7 Master-slave oscillator configuration. 81
6.8 Master-slave oscillator configuration. 82
6.9 Trace showing the internal tension in slave oscillator. 82
6.10 Master-slave oscillator outputs at various values of relative phase. 83
6.11 Master oscillator Simulink model. 84
6.12 Slave oscillator Simulink model. 84
6.13 Master-slave configuration Simulink model. 85

7.1 Control policy setup for learning of rhythmic task. 87
7.2 Partial time series of recovered and reproduced joint angle trajectories. 91
7.3 Recovered and reproduced stick tip trajectories for subject one. 92
7.4 Recovered and reproduced stick tip trajectories for subject two. 93
7.5 Recovered and reproduced stick tip trajectories for subject three. 93

7.6 Frequency analysis of observed and reconstructed drumstick tip trajectory. 95

8.1 Partial time series wrist and fulcrum joint angles for Moller drum stroke. 103

A.1 Results of center of oscillation analysis for the single stroke task, subject one. . . . 111
A.2 Results of frequency of oscillation analysis for the single stroke task, subject one. . 112
A.3 Results of amplitude of oscillation analysis for the single stroke task, subject one. . 113
A.4 Results of relative phase analysis for the single stroke task, subject one. 114
A.5 Results of center of oscillation analysis for the single stroke task, subject two. . . . 115
A.6 Results of frequency of oscillation analysis for the single stroke task, subject two. . 116
A.7 Results of amplitude of oscillation analysis for the single stroke task, subject two. . 117
A.8 Results of relative phase analysis for the single stroke task, subject two. 118
A.9 Results of center of oscillation analysis for the single stroke task, subject three. . . 119
A.10 Results of frequency of oscillation analysis for the single stroke task, subject three. 120
A.11 Results of amplitude of oscillation analysis for the single stroke task, subject three. 121
A.12 Results of relative phase analysis for the single stroke task, subject three. 122

B.1 Partial time series of recovered and reproduced joint angle trajectories. 125
B.2 Partial time series of recovered and reproduced joint angle trajectories. 127
B.3 Partial time series of recovered and reproduced joint angle trajectories. 129

List of Tables

3.1 Results of accuracy and precision studies of Motion Analysis system. 26
3.2 Kistler force plate properties as per manufacturers specifications. 26
3.3 Flexiforce force sensitive resistor properties as per manufacturers specifications. . . 27
3.4 Location and configuration of real markers. 30
3.5 Trials stroke type and single hand playing frequencies performed by each subject. . 32

4.1 Body model segment definitions. 45
4.2 Body model joint center virtual marker definitions. 47
4.3 Resulting joint center virtual maker offsets. 54
4.4 Joint center virtual maker offsets used in remainder of work. 55
4.5 Resulting segment lengths and joint center residual errors, all units in mm. 58
4.6 Magnitude of standard deviation of impact coordinates (X,Y). 62

7.1 Oscillator parameters resulting from learning task. 90
7.2 Resulting segment lengths, and joint center residual errors. 92
7.3 Frequency, amplitude, and phase of oscillation of drumstick tip trajectory. 93

List of Symbols

DOF	Degree of freedom
CPG	Central pattern generator
CCD	Charge-coupled device
MSE	Mean squared error
LSQ	Least squares
bpm	beats per minute

List of Terms

Flexion	act of bending a joint so that the angle between the two limbs is decreased.
Extension	act of stretching or straightening out a flexed limb.
Abduction	moving of a body part away from the central axis of the body.
Adduction	moving of a body part toward the central axis of the body.
Medial	relating to or situated in or extending toward the middle.
Lateral	lying away from the median and sagittal plane of a body.
Pronation	rotation of the hands and forearms so that the palms face downward.
Supination	rotation of the hands and forearms so that the palms face upward.

Chapter 1

Introduction

1.1 Overview

Humanoid robots are emerging as a topic of interest in the field of robotics, and have spawned several related topics of research. Two of these topics which have recently received considerable interest are motion control, and learning for humanoid robots. A challenge in humanoid robot control is producing organized, purposeful movements, as well as human-like movements, because humanoid robots are highly redundant, multi-degree of freedom robotic manipulators. Robot learning and behavior has been largely limited to specific, preprogrammed tasks. Analyzing and programming behavior for specific tasks and conditions is time consuming, and must be done by someone with a great deal of knowledge about both robotics, and the task. This approach also represents an unnatural, unintuitive interface for interacting with and teaching humanoid robots. A new area of research that is proving to be successful in addressing both of these problems, and is emerging as a powerful tool for robot learning, is robot learning by imitation. Imitation learning has thus far allowed researchers to enable humanoid robots to learn to juggle, balance a pole, and learn the peg-in-hole game [9], dance and catch [73], throw [47], learn a tennis forehand and backhand swing [77], hand drumming [33], and learn to play 'sticky hands' [30]

This work pursues robot learning from an imitation approach; specifically, visually mediated imitation learning of rhythmic motor tasks. The rhythmic task used for this work was drumming, and was performed by three expert subjects. The movements were recorded using a Motion Analysis passive optical motion capture system. The captured data was processed to extract kinematic data (joint centers and angles,and segment length) that could be used to control a humanoid robot. This step also included constructing a model of the humanoid robot that could be imposed on the motion capture subject. The motion was then learned by robot using a simple direct-policy learning framework, allowing the robot to reproduce the demonstrated motion. The learning by imitation framework is a powerful robot learning tool that allows humanoid robots to easily learn a wide variety of movement behaviors which would otherwise be difficult to teach a robot.

1.2 Approach and Direction

The first step in an imitation learning framework is quantitative measurement of motion to be learned. Many methods of motion measurement have been used for imitation learning; some are costly commercial systems originally created for biomedical gait analysis, and some that were developed by individual research labs specifically for the purpose of imitation learning. For this work, a state-of-the art Motion Analysis commercial passive optical motion capture system was available, and thus was used. This is a very powerful and accurate motion capture system and will be discussed in chapter 3. In most cases, a required intermediate step for imitation learning to be possible is that the captured motion (regardless of what method was used for capturing it) must be transformed to a form that is compatible with robot control parameters, such as kinematics data. This step itself involves several issues, which depend on the method of motion capture, and will be discussed in chapter 4. Because an optical motion capture system was used to capture motion, this work focuses on methods for obtaining kinematic robot data that are specific to optical motion capture systems. For the learned behavior to be encoded and stored for later recall, a method of representing it must exist. This learned representation of motion is often called a control policy or motion primitive. For this work, the popular Matsuoka neural oscillator was chosen for two reasons: first, as it is a dynamic system, its state has no explicit dependency on time. As will be discussed, this is a desirable property for a control policy. Second, this particular oscillator has received much attention in robotics, and much analysis of its properties and behavior has been done. The actual learning strategy used in this work was a simple one, which on its own is not very powerful, but is better suited as the fist step of other more powerful learning strategies.

As a demonstration task, or task to be learned, three distinct drumming strokes were used. This task was chosen for a number of reasons: first, it is a rhythmic task with a reasonable degree of precision, and employs a tool that interacts with a fixed surface. Drumming also presents some characteristics other than 3-D or joint trajectories (such as joint impedance modulation) that would be interesting to incorporate into an imitation learning framework.

Scope

Due to time restrictions, many simplifications were made to this work, and many of the original goals were modified or eliminated altogether. First, only the kinematics of motion were considered; dynamics were not. This simplification in turn required that the original goal of joint impedance learning be eliminated. As no humanoid robots were available for implementation, a kinematic simulation of the robot was created using Matlab and thus the presented results are only theoretical. Additionally, although three types of drumming strokes were to be used as learning tasks, learning was performed on only one of the strokes. As a result of this, much of the captured motion data, including two strokes, impact force recordings for all tasks, and muscle activation levels (using EMG's) of some tasks have remain unused. Not all issues related to imitation learning were

addressed it this work. Attention mechanisms, movement classification, mismatch of teacher and imitator bodies, and learning of sequence of movements were not considered. Finally, the learning strategy employed was very simple since dynamic characteristics of the task were not considered.

1.3 Thesis outline

Chapter 2 will review other research undertaken in imitation learning specifically, as well as related aspects including perception and measurement of motion, extracting kinematic data from optical motion capture data, control policies and motion primitives, and imitation learning modes, and various approaches to imitation learning in general.

Chapter 3 will detail the methods and equipment used to measure human motion in this work, and describe the types of motion that were captured, and the subjects that were used.

Chapter 4 will describe the methods used for extracting usable robot kinematic data from the 3-D motion capture data, including the body model used, and joint center and angle, and segment length estimation methods. The accuracy of the system will also examined, and results of this presented and compared to the accuracy displayed by the human subjects in the same tasks.

Chapter 5 will analyze one type of motion capture trial, and attempted to highlight visible trends and preferred values across the different trial conditions and subjects.

Chapter 6 will examine properties of the Matsuoka neural oscillator, and its applications as a motion primitive, and present the oscillator configuration used in this work.

Chapter 7 will present the imitation learning procedure and results for one type of task, including the learning mode and objective function, and again compare the results of learning and reproducing the task to the abilities of the subjects for the same task.

Chapter 8 will conclude the thesis, and suggest small improvements that can be made to the described system to improve its learning abilities, as well as provide suggestions for future work.

Appendices A and **B** contain figures showing the results of the task analysis (chapter 5) and direct policy learning (chapter7) respectively.

1.4 Contributions

The contributions of this thesis are as follows:

1. In contrast to previous approaches, which have relied on joint angles sensors to record human rhythmic motion, this work employs an highly accurate optical motion capture system to track key features of interest on the human performer. The advantage this represent over other work is two-fold: First, the employed method is capable of capturing motion with a much higher fidelity; more degrees of freedom can be captured with more accuracy and at a faster sampling rate than previous methods. This translates a more complete, faithful representation, and subsequently to better learning of the behavior. Second, the employed motion capture system expresses motion in 3D Cartesian

coordinates (in contrast to joint angle coordinates), which allows for direct comparison between the observed and recreated motion in Cartesian coordinates. This direct comparison is necessary to quantify the accuracy with which a behavior was learned using goal-related variables (such as the end-effector trajectory) instead of simply joint angles.

2. This work attempts to quantify the performance of the described imitation learning model by proposing two suitable metrics (one specifically for rhythmic motion), and presenting the results of applying the metrics to the learned drumming task.

3. This work presents results suggesting the popular Matsuoka oscillator is not a good model of human rhythmic motion, and as a result highlights an important new characteristic that a primitive for rhythmic motion should possess.

4. This work develops and imitation learning model, and through implementation shows it can be use to teach robots rhythmic tasks involving several degrees of freedom and complex phase requirements between the degrees of freedom.

Chapter 2

Related Work

This chapter will introduce the main issues in robot imitation learning, and review relevant literature which addresses each of these issues. In general, only visually mediated imitation of motor tasks has been considered in robot imitation learning, thus many of the issues in imitation learning relate to problems surrounding the capture and processing of motion data. The major components of a robot imitation learning framework are summarized below:

- Movement perception and capture
- Change of frame of reference of motion data
- Attention focus mechanisms
- Matching of imitator and teacher kinematics
- Motion primitives (or, control policies) for representation of motion
- Types of imitation learning
- Test-beds for imitation learning experiments

2.1 Visual Perception of Motion

2.1.1 Motion perception and Capture

In order to recognize, understand, and subsequently learn a task from a human demonstration, the robot must first be able to perceive motion. This captured movement data is what the robot will use to learn a new behavior. The perception task is in essence an object recognition and tracking problem. This can include tracking of a subject as a whole, as well as tracking the movement of individual limbs relative to the body. Depending of the type of system, motion is captured as a set of 3-D coordinates of points of interest on a subject, or a set of joint rotations and segment lengths corresponding to a model of the subject whose motion is to be captured. The systems or methods used to capture motion (human or otherwise) fall into four main categories: optical, electromagnetic,

mechanical/goniometer, and computer-vision based systems. Following is a brief description of each, and a summary of how they have been used in imitation learning research.

Optical motion capture systems

Optical motion capture systems capture human motion by recording the positions of markers placed on the subject's body. The position of each marker is recorded by at least two high-speed cameras with infrared-pass filters. The markers can be either passive, or active: Passive markers are small (1 to 3 cm) balls covered with retro-reflective tape. The infrared source is an array of infrared LED's, usually located on each camera around the lens. The infrared emitted by the arrays light is reflected from the markers and recorded by the cameras. Active markers in contrast, are pulsed infrared LED's. Data from each camera is then combined using various proprietary methods to calculate the marker's coordinates in 3-D space. At least two cameras are necessary to capture 3-D motion; most modern systems use at least four cameras to provide larger capture volumes and to reduce occlusion of markers. Sampling rates for these systems can be up to 300 Hz. Motion Analysis Eagle [4] in an example of such a system.

For accurate data, these systems require calibration of the cameras. Since the markers can be placed on nearly any surface, motion capture of objects other than human subjects is possible. The accuracy of optical motion capture systems is in the millimeter range or better [72], and is dependent on the calibration quality. Integration and synchronization of other data sources are well supported. A notable problem with these systems is marker occlusion and swapping. Occlusion occurs when an objects obstruct the visibility of a marker by the cameras. Swapping occurs when the trajectories of two or more markers are swapped when they are close to one another. Both problems lead to noisy or incomplete data that must be corrected during post processing. Due to their prohibitive costs, this motion capture technique has not been used significantly in imitation learning research.

Riley et al. [73] and Ude et al. [85] used an OptoTrak system with three stereo cameras to record data from 22 markers attached to subjects and sampled at 60 Hz to capture motion of karate and Japanese folk dance sequences. Miyamoto and Kawato [55] also used an OptoTrak system to record motion of a tennis serve at 250 Hz. Markers were placed on the subjects shoulder, elbow, and wrist, and upper and lower sides of the tennis racket.

Mechanical or goniometer based motion capture systems

Goniometer based and mechanical motion capture systems generally consist of a full body suit worn by the performer, to which an exoskeleton is connected. The exoskeleton is made of a set of lightweight rods connecting each of the performers joints. At each joint, an angular-position measuring device, such as a potentiometer, an optical encoder, or hall effect sensor measures the angle between the two connecting rods. This data is interpreted using a calibration file unique to each performer to calculate the actual angles of each of the the performers joints. Calibration files

typically relate the offsets from the sensors rotational center to the performers joint center, and the length of each segment or limb calculated as the distance between joint centers. To determine the bearing (rotational direction) of the performer, some systems are equipped with a gyroscope mounted on the performer. On high end systems, most or all of the processing (joint angle, segment length , and orientation, digitizing) is done on the suit itself, and data is sent to a host computer for recording and post processing. This gives some goniometer based systems a larger capture volume than most other systems. The maximum sampling rate of goniometer based systems is in the area of 120 Hz; somewhat slower than optical systems. The Gypsy, by MetaMotion [59] and Sarcos Sensuit [75] are examples of such systems.

Goniometer based systems have the added advantage that they measure human motion directly in joint space, and do not suffer from marker occlusions as do optical systems. Goniometer based systems are not flexible in the type of motion they can capture however, as the kinematic structure of the subject is fixed, and defined by the suit. Motion capture of subject other than humans, or motion capture with tools (such as sports equipment) are not facilitated or impossible, as the suits are designed to record motion of the human body only. Integration and synchronization of other data sources such as force plates is not supported. Also, some researchers have reported problems with the suits not fitting persons of extreme size [85]. Finally, assessment of the accuracy of these systems is not readily available.

Ijspeert et al. [34][33] directly recorded joint angles of subjects performing reaching and rhythmic movements using the Sarcos Sensuit. Six degrees of freedom (DOF's) for both arms were recorded (three at the shoulder, one at the elbow, and two at the wrist) at 100 Hz. Schaal [77] also used a Sarcos Sensuit to record tennis forehand and backhand demonstrations. Matarić [48] also used this system for human motion measurement.

Magnetic motion capture systems

Magnetic motion capture systems measure the signal of electromagnetic sensors in a magnetic field, and interpret this data to determine the 3-D position and orientation of each marker. Sensors are typically tethered to a transmitter (though untethered models are available), reducing allowable motion. The number of markers are determined by the number of sensor inputs on the transmitter (or radio-pack for untethered models). The transmitter also determines the capture volume, with the maximum being about a 5 m radius from the transmitter. Effective sampling rate of electromagnetic systems is in the range of 120 Hz. Skill Technologies 6DR StarTrak [35] is an example of a magnetic system. Magnetic systems have several accuracy related problems: they are susceptible to interference from metallic objects (such as tools, metal in floors and walls) within the magnetic field [68][72][54], their accuracy decreases as the distance from the transmitter increases [72], and the sampling rate affects the systems accuracy in the presence of metallic objects [45].

In [48][12] Billard and Matarić had subjects watch and then imitate short videos of arm move-

ments, while wearing the FastTrak marker system for recording the 3-D positions of three markers on the arm: near the shoulder, near the elbow, the wrist, and the hand.

Computer-vision based motion capture systems

In contrast to commercially available motion capture systems which require markers or sensors to be placed directly on the subjects body, purely computer-vision based motion capture systems represent a non-intrusive alternative method of motion capture, and are gaining popularity among researchers as an affordable alternative to expensive commercial systems. No computer-vision based motion capture systems are commercially available, and thus researches generally develop systems from scratch specifically to suit the needs or their research.

Bottino and Laurentini [14] capture human motion by first constructing a 3-D volumetric description of a performer by volume intersection of 2-D silhouettes obtained from several cameras. Background subtraction and back-projection are used to obtain the 2-D silhouettes. Human posture is then determined by using a human model with 15 segments and 32 DOF's, and finding the joint angles that minimize the distance of each segment to an oriented bounding ellipsoid used to represent each of the human segments constructing from the silhouette intersection. However, no statement about the systems accuracy in measuring live human motion were made in the work. D'Appuzo [19] uses a squared error (LSE) matching algorithm to extract 2-D coordinates of specified features of the performer. The 3-D coordinates of these points are then calculated from the 2-D data by forward ray intersection using the orientation and calibration data of the cameras. This approach is similar to that of optical motion capture systems, the difference being this method calculates the 2-D position (with reference to each camera) of body features instead of retro reflective markers. D'Appuzo's approach however, does not handle occlusion of the key tracking points, and requires considerable user intervention. Also, though results for simple upper-body motion were presented; no mention is made of the systems ability to track full body motion, or of its accuracy. Chu et al. [17] propose a model free, marker-less 3D motion capture system for articulated kinematic structures based on non-linear spherical shells (NSS). This approach is centered on the NSS method for generating underlying nonlinear axes (or a skeleton curve) from the volume of an arbitrary rigid-body model, and thus extracting a skeleton curve feature from a Euclidean-space volume of points. In [37][47][48][10][12][11] a 2D motion tracking system is used that extracts features from a video stream. Features were considered to be fixed positions on the body of the subject. The system relied on a constrained (un-occluded and unambiguous) initial position and kinematic model of the object or body being imitated. It calculates the positions (relative to a fixed, extrinsic frame of reference) of nine points on the body: one on each wrist, elbow, and shoulder, and one on the lower torso, neck head, to provide a stick figure of the subject's upper body in 2D. Tracking is done off-line and based on image frequency of 15 Hz. A kinematic model of generic human adult is used in the work, which restricted the capture to human subjects. In a comprehensive survey of computer-vision based

motion capture systems by Moeslund and Granum [57] conclude the field is still in an early stage of development: current systems make several assumptions to simplify tasks, therefore limiting their applications. Many systems posses a low level of automation, especially in the initialization stage. Varying motivations for computer-vision based motion capture systems have produced systems with widely varying degrees of robustness, accuracy and speed. For these and other reasons, the authors conclude this technique is not yet mature enough to be used for general, reliable, accurate motion capture.

2.1.2 Attention Focus

A more recent trend in imitation learning takes a biologically inspired approach, and uses neuroscience and biological data as a basis for developing imitation learning architectures and algorithms that are more general and less task specific. This vein of research has highlight an attention module component as one that is necessary for autonomous, real-time imitation learning. For this type of learning, a robot must be able to select relevant information from the perceived motion stream, and automatically focus attention on features that are salient to the task. Humans do this by social context either by gaze direction, pointing, verbal and postural cues and gestures, or knowledge of the task to learn [1]. Autonomous, real-time robot imitation learning architectures must also posses this capability. Notably, this component of an imitation learning framework only applies to computer-vision based motion capture, since with the other discussed systems, the feature of importance is implied by the markers.

Adams et al. [1] have implemented an attention mechanism comprised of motion and color detectors, and a motivational and behavioral model, which endows a robot with exploratory behaviors toward objects in the environment that have inherent or contextual saliency. Their implementation allowed them to demonstrate preferential looking by a robot. Experiment by Matarić and Pomplun [49], in which 40 subjects were asked to wear an eye-tracker, and watch and imitate videos of finger, hand, and arm movements, showed that subjects consistently gave more attention to hand or end-effector tools for several demonstration tasks. The authors thus suggest that people analyze human arm movements largely by tracking the hand or the end-point, even if the movement is performed with the entire arm. Based on the results of this experiment, Matarić [48] and Jenkins et al. [37] develop a selective attention mechanism for extracting salient information form visual stream, by focusing on the subjects' end effector (hand or tool). The system performs segmentation in retinal space, evaluates current interest level of each feature, and places center of attention at the area with highest interest level. Interest level is determined based upon three factors: retinal movement, captivation, and task level bias. Retinal movement increases attention to features that display motion. Captivation is based on recent interest levels, and decrease interest to features that have previously been given attention, but produce little or no new motion. Finally, task-level bias gives preference to specific features involved in a task. In [37], Jenkins concludes that endpoint trajectory track-

ing is enough for simple movements, but not for complex kinematic structures. Billard and Matarić [10][12] incorporate a simplified attentional module into the vision system, which is triggered whenever a significant change of position (relative to the position at the previous time step) in one of the limbs is observed. This is implemented as a low-pass filter on the calculated joint angle velocities. The attention module also creates an inhibition in learning the module, preventing information flow to learning module, thereby allowing learning of new movements only when a change in the limb position is observed.

2.1.3 Generating robot kinematic data from motion capture data

For learning of demonstrated motion to take place, the example data must be in a form that is suitable for kinematic robot control; namely, joint angle position or velocity. Motion capture data is usually in the form of a set of 2D or 3-D coordinates in Cartesian space, in an extrinsic frame of reference, of the features of interest to track (e.g. optical or magnetic markers). Generating usable data for robot control from motion capture data involves relating the 3-D coordinates of the features of interest (in an extrinsic frame of reference) to the underlying joint angles of each DOF (in intrinsic frame of reference). This problem has been examined under several names, including "pose estimation", "coordinate transform", and "generating kinematic from motion capture data", and contains several distinct sub-problems depending not only on the area of research, but in the method of motion capture itself. Though several methods exist to accomplish this, they all have some common features. Underlying to all the methods is a body model of the object whose motion is to be captured. This model is in the form of a set of articulations connected by rigid segments. Articulations represent joints, and segments represent rigid body parts, usually underlined by a bone. The body model topology is defined by a hierarchy tree and may be of any shape which allows for tree-like representation of the connected articulations. Most often the motion capture subject is human. In this case, there is no agreement among different works in the level of complexity required by the model in terms of number of segments and DOF to properly capture human motion, particularly with regard to the shoulder and spine region. This problem has been examined widely in computer graphics applications, where recorded data is used to animate a virtual characters. Commercial systems are available that accomplish this task, but they generally rely on a proprietary 3-D data format associated with the motion capture system, and are costly. Finally, motion capture with goniometer based systems is not subject to this problem, as the data is captured directly in joint space.

Bottino and Laurentini[14], and Chu et al.[17] accomplish this directly in the tracking phase, though they use different techniques. Bottino and Laurentini first obtain a 3-D volume of the subjects body from silhouette intersection of several cameras. The then fit a human body model (15 segments, 32 DOF's) to the volume data, thus recovering the underlying joint angles. Chu et al. use a model-less approach, which automatically derives a model of the subjects body. Chu et al. first remove pose-dependent non-linearities (caused by rotations about joints) from the volume, by trans-

forming the volume into an intrinsic space. The then cluster the pose-independent volume to highlight the principal curves (in intrinsic space), and finally project points defining the intrinsic space principal curve into the original Euclidean space to produce a skeleton curve for the volume, thus recovering the underlying joint angles. A similar silhouette approach is also used in [37][47][48] [10][12][11], but only upper body motion is tracked. Furthermore, only a 2D silhouette of the subject is created, which makes obtaining joint angles from feature coordinates trivial. Billard[10] and Billard and Matarić[12] mention transforming the Cartesian coordinates of each joint of the demonstrator's limbs in an extrinsic frame of reference, into an intrinsic frame of reference relative to the demonstrator's body, by calculating the joint angles of the elbows and shoulders. However, the do not mention how this is accomplished. Pollard et al.[66] map raw marker data onto a skeleton having 39 DOF's in the upper body, using Vicon's Bodybuilder software.

Riley et al.[73] and Ude et al.[85] generate kinematic data for an existing humanoid robot, and consequently specify a body model that coincides with the kinematic structure of their robot which had 30 DOF's and 24 segments. The kinematic data is generated from motion capture data (obtained using an OptoTrak system) by first modeling the locations of the markers with respect to the underlying skeleton model in an initialization step, and then estimating the joint angles (through LSE minimization) in each frame of data that minimize the difference between the measured and reconstructed marker 3-D coordinates. A similar approach is used in [82][23][24][22][15], but a slightly different intrinsic frame of reference is defined. These works also use a much more complex body model (with 62 DOF's) since the application is realistic reproduction of human motion for computer graphics applications, not robotics. Furthermore, this same framework is also used in [58][31] to improve the performance of marker based tracking. Rab et al.[71] define an upper body model consisting of 10 segments (head, neck, shoulder girdle, left upper arm, right upper arm, left lower arm, right lower arm, left hand, right hand, pelvis) with 18 DOF's. Joint angles are obtained by creating coordinate systems at each joint (requiring three non-collinear markers attached to each rigid segment), and calculating the rotations along each axis between a segment and its parent segment. Rab et al. also provide results on the accuracy and robustness of their methods in recovering underlying joint angles; something missing from most other works. Choi et al. [16] employ a similar technique, but re-adjust the recovered joint angles at several key-frames using inverse kinematics to ensure the recovered and measured endpoint coordinates are consistent. Similarly, Zordan and Hodgins [91] use inverse kinematics to re-target the position of the hand while still maintaining its orientation, by modifying preceding DOF's (wrist, elbow, shoulder). Zordan and Hodgins also use dynamic simulation constraints to re-target points where contact with a fixed object is made.

2.1.4 Correspondence of kinematics

In some instances, the kinematic structure of teacher and subject do not match. The mismatch can be in the number, organization, range, and speed capabilities of the DOF's, or in the lengths of the

segments. This mismatch between the teacher and imitators' bodies is often called the correspondence problem, and it presents a problem in an imitation learning framework for both learning of a new tasks, and evaluation of the success of learning the new tasks. Though most researchers acknowledge the impact of this problem, few have tried to addressed it, and instead work around it by forcing teacher and imitator agents to the same kinematic structure.

Polard et al.[66] first map raw marker data onto a skeleton having 39 DOF's in the upper body (using Vicon Bodybuilder software). The motion capture data is then mapped to the less flexible skeleton of their humanoid robot by setting joint angles to match the orientation of each segment of the robot with the corresponding link or set of links of the human skeleton. Head, upper arm, lower arm, and hand orientations were matched directly, as both skeletons contain those links. The torso orientation of the robot was set so that a triangle containing the waist and shoulder positions had the same orientation for the two skeletons. They use non-uniform, local scaling to modify each trajectory of the motion to lie within the joint limits of the robot. The velocity of the each obtained joint angles is also scaled by averaging the results of an ideal, simulated, velocity-limited tracking controller run both forward and backward in time. Alissandrakis et al.[5] propose a framework for dealing with the correspondence problem called ALICE (Action Learning for Imitation via Correspondences between Embodiments). This methods builds a correspondence library that maps the (structure of) perceived motion by a demonstrator to the attempted match result (given by the generating mechanism used). The best correspondence is the one that minimizes some metric of degree of success of imitation. New correspondences are learned by trial and error. The actual metrics used to evaluate performance can vary, and dramatically affect outcome.

2.2 Control policies for imitation learning

Control policies, also called motion primitives, are a necessary method of representing or encoding an observed motion so it may be subsequently used for robot control, and so learning can take place. Motion primitives encode the few basic types of movements (postural, discrete, and rhythmic), from which other more complex movements are composed. New behaviors are learned by determining the parameters and spatio-temporal combination of primitive instances required to produce the observed behavior, which are then reproduced by activating the appropriate control policies. Thus, primitives are necessary for both learning and performance of new movements, and represent the generic building blocks of motion that can be implemented as parametric motor controllers[47]. Control policies specify desired trajectories,and can do so in joint or Cartesian space. Ijspeert et al.[32] highlight some important desirable properties of motion primitives, which include their:

- ease of representing and learning a goal trajectory
- compactness of representation
- robustness against perturbations and changes in a dynamic environment

- ease of reuse for related tasks and easy modification for new tasks
- ease of categorization for movement recognition

Two of these properties, a primitives robustness and ease of reuse, are related to whether the primitive is autonomous. Schaal et al.[77] define an autonomous control policy as one that is not explicitly time dependent or time indexed, but instead is only state dependent. Several motion primitives have been suggested and used for humanoid robot imitation learning and control, with varying degrees of attention and success.

Recorded trajectories

The simplest motion primitive involves simply recoding the trajectories from a suitable demonstration at a specified interval, and subsequently using the recorded points directly to produce future motion. Though memorized trajectories are easy to learn, they suffer from several significant drawbacks. Because they are time indexed, they are hard to reuse for new tasks, and are not robust toward significant perturbations or changes of the environment. Recorded trajectories can also become extremely large if the duration of the recorded motion is long, the capture rate is high, or many features are tracked. Furthermore, they do not aid in categorization of motion types. This approach is quite limited, and has not received much attention.

In [80], Ude et al. use a PD controller running at 420 Hz to track recorded joint angle trajectories to enable a humanoid robot to copy a set of dancing and karate sequences.

Splines and wavelets

Recorded trajectories can also be represented using splines and wavelets. Although these methods have a more compact representation, and can somewhat facilitate the reuse of behaviors for slightly different tasks, since they are non-autonomous they suffer from some of the same problems as memorized trajectories; they are not robust to perturbations, and do not aid in movement classification. However, their relative ease of use has garnered them considerable attention in imitation learning and robot control research.

Atkeson et al. [9] employed B-spline wavelets for robotic arm to learn a pole balancing task. Miyamoto and Kawato [55] [56] approximated end-point trajectories as splines and via-points, to allow a robotic arm to perform tennis serve, and play the kendama game. The learned behavior was also modified for slightly different tasks by adjusting the via points through time scaling. Hale and Pollick [30] employed piecewise linear splines to allow a humanoid robot to play the game of "sticky hands" with another human.

Dynamic systems

Dynamic systems can also be used to encode recorded trajectories, and have recently become the focus of considerable interest in imitation learning, especially among biologically inspired imitation

learning research. Because their output is not time indexed, dynamic systems are autonomous motion primitives, and thus produce control policies which are robust to strong external perturbations [32] Furthermore, dynamic systems provide an effective method for motion classification. Using dynamic systems as control policies generally requires the tuning of relatively few parameters to modify the control policy topology to match that of the observed behavior, and thus learn the desired behavior.

Billard and Matarić [10][12] employ discrete and oscillatory dynamic systems as control policies to allow an avatar to learn a wide range of full body motions. Ijspeert et al.[34][33] use non-linear oscillators to realize imitation learning of hand drumming and tennis swing tasks. In [2][3], Amit and Matarić suggest the use of dynamical systems as discrete and oscillatory control policies, but leave the selection of dynamic system up to the designer at implementation time, to best match the observed behavior. Williamson[89] uses a particular instance of a dynamic system, the Matsuoka non-linear neural oscillator, to exploit the natural dynamics of a humanoid robot, and thus realize a wide range of rhythmic tasks including sawing, crank-turning, drumming, and playing with a slinky toy.

Other motion primitives that have been used include straight line, circle, and arc trajectories [37].

2.3 Imitation learning modes

Provided all all preceding components in an imitation learning framework are in place, that is, provided the behavior has been observed and recorded, the data has been converted to the appropriate coordinate system or frame of reference, the body correspondence problem has been addressed, and a suitable control policy encoding method has been chosen, the actual learning of the behavior can take place. Learning Involves matching the imitated behavior to the observed behavior. For learning or improvement of the behavior to take place, a criterion function relating the quality of the imitated behavior to the demonstrated one is necessary. Possible definitions of this criterion function, and which specific variables they consider introduces the concept of learning modes. Several learning modes have been suggested, and they differ mainly in the type, or the characteristics of the behavior that is being learned or imitated. Some emphasize strictly the goal to be achieved, while others emphasize the method in which the goal was achieved, while others still try to uncover the underlying dynamics of the task[76]. Consequently, a combination is sometimes necessary for learning to take place[77].

Direct Policy Learning

In direct policy learning, the student has no knowledge of the task goal, and simply copies the observed behavior without knowledge about how to modify it for a new behavioral context. This, type of imitation learning is limited in that it can only learn what it can observe, which is kinematic

data; either in joint angles (internal coordinates) or Cartesian trajectories (task-level, external coordinates). Consequently, criterion functions for direct policy learning are defined in terms of kinematic variables such as joint angles or observed trajectories, and are simply the squared error between the observed and reproduced behavior. When the observed behavior is in task space (i.e. external, or Cartesian coordinates), prior knowledge is required of how a task-level command can be converted into an actuator-level command, and vice-versa (i.e. forward and inverse kinematics). With direct policy learning, the student will not be able to undergo further self-improvement to achieve the task goal unless an explicit goal-level reward signal is provided. Additionally, there is no guarantee that the imitated behavior is stable, or can even can accomplish the desired or implied behavioral goal[77]. The policies generated by this type of learning are poor in that they cannot be re-used for even slightly modified behavioral goals[76]. This learning mode has is often called indiscriminate imitation or mimicking, as it just repeats an observed action pattern without knowledge or the goal[77]. Billard and Matarić[12] employed direct policy learning to enable a human avatar to learn reaching motions based on joint angle trajectories, and oscillatory arm movements ("raise the roof" motion) based on Cartesian shoulder, elbow, and hand trajectories from human demonstrations. In [10], a human avatar also learned a large number of series of movements involving the the shoulders, elbows, hips, knees, fingers and wrist, from another human avatar. Ijspeert et al.[33] enabled a humanoid robot to learn a figure-8 motion traced with the arm and hand drumming patterns on joint angle data, and 2D letter-tracing on Cartesian coordinates. Amit and Matarić[2][3] learned joint angle data of a wide variety of discrete and oscillatory movements involving shoulders, elbows, and hips with both real, and synthetic motion capture data.

Matarić[47] recorded joint angle data of a human dancing the Macarena, and subsequently employed direct policy learning on the joint angle data to allow a simulated human to dance. Riley et al.[73], Ude et al.[85], and Pollard et al.[66] also employed direct policy learning on joint angle data to allow a humanoid robot lo learn an okinawan folk dance, karate, and "I'm a little tea pot" sequences. Direct policy learning was also used by Jenkins et al.[37] to learn various discrete and oscillatory motion end-effector trajectories.

Goal directed policy learning from demonstrated trajectories

A more powerful learning mode, goal directed policy uses recoded trajectories (joint angle or Cartesian) as a seed for an initial policy to be optimized by a goal-level based self-improvement process. An important difference with respect to direct policy learning, is that now the the task goal is known, and used in the self-learning optimization process. This approach results in policies that attempt to reproduce the task goal, but still preserve the original movements of the demonstrator. Furthermore the self-learning adjusts the imitated movement to kinematic and dynamic discrepancies between the student and the teacher, and ensures behavioral stability [76].

Ijspeert et al.[34] enabled a humanoid robot to learn a tennis forehand and backhand swing on

joint angle data, but did not specify what goal-specific criteria was used. Miyamoto et al.[56] used this type of learning to allow a 7 DOF arm to learn the kendama "ball in cup" game. First the trajectory of the ball was captured using an optical motion capture system. the necessary joint trajectories were obtained using inverse kinematics. To correct for kinematic and dynamic differences between the demonstrator and robotic arm, and thus allow the robot to successfully play the kendama game, the joint angle trajectories were modified using goal-specific variables, which were the ball trajectory and cup coordinates. Miyamoto and Kawato [55] employed this type of learning to allow a 7 DOF robotic arm to learn a tennis throw-up and serve. They first recoded racket coordinate and orientation, then used inverse kinematics to obtain the required joint angles for the robot arm to duplicate the racket trajectory. The joint angle data was then adjusted by self improvement based on task-level criteria (final ball trajectory) to adjust for kinematic and dynamic differences between the demonstrator and robot arm.

Model-based policy learning

Yet another learning mode, model-based policy learning differs from the previous two methods in that the task dynamics are considered. From the observed behavior a predictive model of the task dynamics is approximated. Given knowledge of the task goal, the task-level policy of the movement primitive can be computed with reinforcement learning procedures based on the learned dynamic model.

Atkeson et al. [9] enable a robotic arm to learn how to balance a pole on a finger tip in a single trial, and learn a pendulum swing-up task in a few trials using this approach.

2.4 Imitation learning testbeds

As a tool for evaluating imitation learning experiments, both real humanoid robots and manipulators, and humanoid simulations have been used. Due to prohibitive cost of humanoid robots, few labs have access to them, and consequently, most employ simulations to evaluate imitation learning work.

2.4.1 Humanoid robots and manipulators

DB humanoid robot: The humanoid robot DB, manufactured by SARCOS[75], is part of the Kawato Dynamic Brain Project, at the Japan Science and Technology Corporation[80][81]. DB is a 1.9 m, hydraulically actuated humanoid robot, with a total of 30 DOF's (three at the head, seven at each arm, three at the trunk, three at each leg, and two at each eye). DB has been used a great deal in this vain or research (see [34][33] [9][30][77][76][73] for examples).

COG humanoid robot: COG is a 22 DOF humanoid robot, with two 6 DOF arms, a 3 DOF torso, a 4 DOF neck, and 3 DOF combined in its eyes. COG resides at the Humanoid Robotics Group, in the Artificial intelligence lab at MIT [64]. Though impressive, COG has been used very little in imitation learning (see [1] for examples).

SARCOS dexterous arm: This robotic arm, manufactured by SARCOS[75], is a 7 DOF manipulator, configured to have a similar kinematic structure as the human arm, and resides at the Department of Humanoid Robotics and Computational Neuroscience in Japan (see [55][56][9] for examples).

2.4.2 Simulations

Adonis: Developed at the Georgia Institute of Technology Animation Lab, Adonis is a 3-D rigid-body simulation of a human with full dynamics. Mass and moment-of-inertia information, as well as gravity and external forces are included in the simulation, though explicit balance requirements are not. Adonis consists of eight rigid links connected with revolute joints of one DOF and three DOF's, totaling 20 DOF's, and joint limits and self-collision detection have been added. This test bed is used at the University of Southern California's Interaction lab [61] (see [37][47] for examples).

Physics-based simulator test-bed: Also used at the Interaction lab in the University of Southern California is a physics-based simulator test-bed[63]. The test-bed relies on a number of commercial libraries for modeling, computations, and rendering, and provides gravity as well as collision detection for all objects in the environment. The humanoid has actuated joints from the waist up, totaling 20 active DOF's. The lower body is not actuated, and the character is firmly attached to the ground at the feet (see [2][3] for examples).

Cosmir humanoid avatar: Developed at the University of Dortmund, The humanoid avatar is another testbed used at the Interaction lab in the University of Southern California[62]. The Avatar has a total of 65 DOF's: Shoulders, hips, wrists, ankles and head have 3 DOF's. Elbows and knees have 1 DOF. The trunk is made of three segments with 2 DOF's each. The thumbs have 2 joints, the rest of the fingers have 3. All limbs are attached by hinge joints. The external force applied to each joint is gravity. Balance is handled by supporting the hips; ground contact is not modeled. There is no collision avoidance module. Avatars are used as demonstrators, as well as imitators (see [10][12] for examples).

2.5 Biological Inspiration for imitation learning

Though some researchers have taken a purely computational approach, others try to validate robot imitation learning by taking a biologically inspired approach, and incorporating findings from neurobiology, into frameworks for imitation learning. The following components have been included as part of biologically inspired robot imitation learning frameworks.

Mirror neurons as the link between perception and action

Studies conducted on monkeys have shown a group of neurons located in the premotor cortex area of the monkeys brain to activate both when monkeys perform certain tasks involving arms and hands (such as reaching for objects and bringing them to the body), as well as when they observe another monkey or human performing the same or a similar task. Evidence also exists that suggests a similar

mirror neuron system exists in humans. This neuron activation has only been demonstrated for simple movements involving the arms and hands, and it remains to be shown if the same behavior is observed for movements of other limbs. This evidence points to the mirror neural system as the system responsible for the linkage of self-generated and observed actions. Whether mirror neurons are actually part of imitation system (in monkeys or humans), however, has remained speculative so far. None the less, mirror neurons have been interpreted by some robot imitation learning researchers as the unifying link between action and perception, in that they may be responsible for matching the neural command for an action with the neural code for the perception of the same action executed by another primate. The mirror neural system findings have been embraced by some robot imitation learning researchers, who have attempted to include an embodiment of mirror neurons into imitation learning frameworks [10][12][48].

Central pattern generators for motor control

There is a great deal of evidence in biological research for the existence of postural, discreet, and rhythmic central pattern generators (CPG's) in various animals, and insects (including cat walking, lamprey swimming, locust flight, and cockroach walking), as well as primates (including humans). CPG's (also called spinal fields) are located in the spinal cord, and produce basic movements which can be modulated though perceptual feedback to produce useful motions. CPG's correspond to motion primitives in an imitation learning framework, and are usually implemented as dynamic systems. They and have been used extensively in mobile and humanoid robotics. Many imitation leaning researchers have included this component withing the framework of a biologically inspired imitation learning model [1][37][10] [12][47][48][2] [3][26]

Chapter 3

Human Motion Measurement

3.1 Introduction

This Chapter will describe the methods and equipment used to measure human motion, and the types of motion and subjects that used for this work. First, existing types of human motion measurement devices (generally called motion capture systems) will be briefly introduced, and their advantages and disadvantage mentioned. Next, an overview of the motion capture system used in this work will be given, focusing on the specific features and components used, and an attempt will be made to quantify the accuracy and precision the system is capable of. Finally, the types of motion captured, their relevance, and subject's expertise at performing them will be over viewed.

3.2 Motion Capture Systems

Though all methods have their advantages and disadvantages, for a particular motion capture application one method if often more appropriate than others. For this application, the motion to capture was contained, and in a small volume. Since the motion was relatively small scale and quick, high accuracy and capture rate were important factors. Also important was the systems ability to capture motion of tools with little interference to their movement, and its ability to integrate with external devices such as analog data sources. Optical motion capture systems display the best performance in most of these features, and thus were determined to be the most appropriate method for this particular application.

3.3 Experimental equipment

Due to the high cost of some of the equipment used for research, the researcher is not always in a position to choose the equipment he or she wishes to work with, and instead must make due with what is available. When selecting a motion capture system to use for this work, this situation was encountered: Having ruled out purely computer vision-based systems for the reasons mentioned in 2.1.1, commercially available systems were the only remaining choice. At approximately $30,000

for a single subject, Goniometer-based systems are the least expensive of the commercially available motion capture systems. Since the ART laboratory at the University of Alberta where this research was conducted is not equipped with a motion capture system, and purchasing one solely for this research project was not feasible, it was necessary to seek out currently installed systems to use for the research. The methods used to generate kinematic data from motion capture systems are not the same for all types of motion capture systems. Consequently, the system eventually used in part determined the direction of the research. In March of 2002, a Motion Analysis Eagle Digital passive optical motion capture system was purchased by the Syncrude motion and balance laboratory in Glenrose rehabilitation hospital. The supervisors of the system were generous enough to grant the researcher access to the system for this research. Following is a description of the system, specific features that were used, and its performance characteristics. Additional equipment used is also discussed. Finally a summary of the complete experimental rig is given.

3.3.1 Motion Analysis Passive Optical Motion Capture System

As mentioned previously, an Eagle Digital passive marker optical motion capture system by Motion Analysis Corporation [4] was purchased by the Glenrose Rehabilitation Hospital for its Motion and Balance Laboratory. The system consists of eight Eagle high speed digital cameras, a hub for both power and data communication to the cameras, passive markers, a 128 channel, 16 bit analog data acquisition module, and Eagle real-time software (EVaRT) from which to control the entire system. Following is a brief description of the components of the system that were relevant to this work. Hardware and software components are described separately.

Hardware

Cameras

The cameras, shown in Fig. 3.1, are equipped with infrared pass filters, and a bank of infrared LED's located around the camera lens. The cameras have a resolution of 1280 x 1024 pixels, and were fixed in the arrangement shown in Fig. 3.2. The cameras are capable of up a 300 Hz capture rate. However, some of the cameras periodically lost synchronization at this rate, and corrupted the captured data, rendering it unusable. The highest functional sampling rate was found to be 240 Hz.

Passive markers

Two sizes of passive motion capture markers are supplied with the Eagle system: 9 mm and 12 mm markers. The 9 mm markers are typically used in clinical gate analysis trials, whereas the larger ones are typically used for motion picture, sports, and video game applications where increased visibility is required. The 12 mm markers have a 2 cm long stem extending from the point of attachment to the subject, to the reflective sphere (i.e. the actual marker portion). This has the effect of magnifying effects of noise due to skin deformation, and reduces the overall accuracy of the system. The point

Figure 3.1: Motion Analysis Eagle digital motion capture camera.

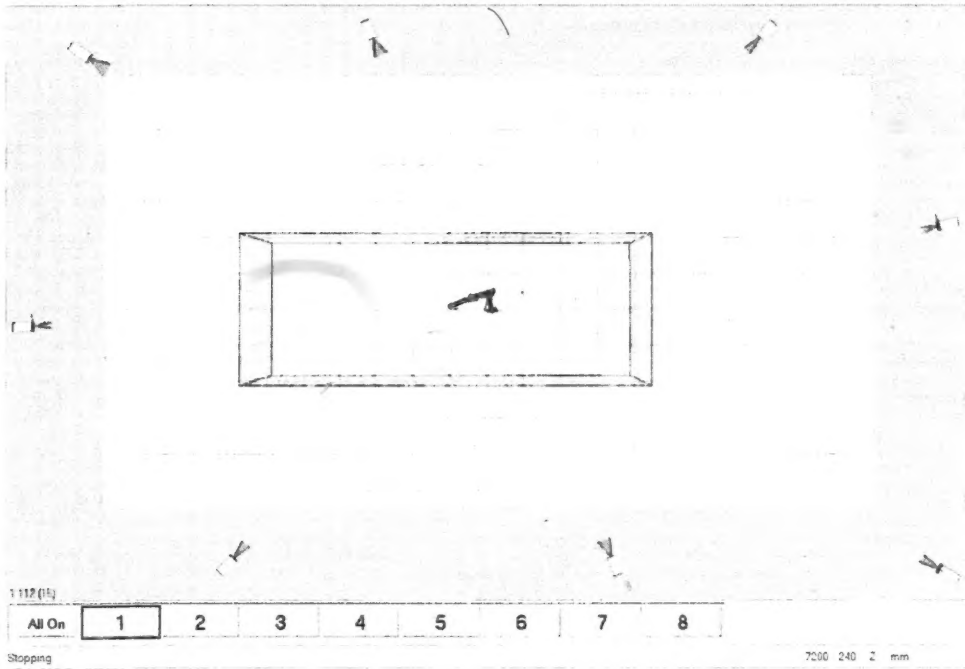


Figure 3.2: Configuration of the eight-camera motion capture system at Glenrose Hospital. The shaded region represents the capture volume, with the subject at its center.

of attachment of these markers also has a large base, and is in fact too large to place on some of the required body parts such as the hands. For these reasons, the smaller 9 mm markers supplied with the system were used for this research with the four exceptions: For the thumb marker, a 9 mm plastic decorative bead was sewn to a small piece of Velcro with adhesive backing, and covered with retro-reflective tape. To make the drumstick tip marker, the drumstick tip was simply covered with retro-reflective tape; the tip was close in size and shape to the 9 mm markers, so the system had no difficulty in tracking it. The drumstick shoulder marker was made by placing a 9 mm wide piece of retro-reflective tape around the stick. this produces a cylindrical instead of sphere shaped marker. Though the systems ability to accurately measure the position of this marker was less than that of spherical markers, for reasons mentioned later, this fact was of no consequence. For the drumstick butt marker, a 9 mm bead was attached to the butt of the stick with a small nail, and covered in retro-reflective tape. Examples of each of the markers are shown in Fig. 3.3.

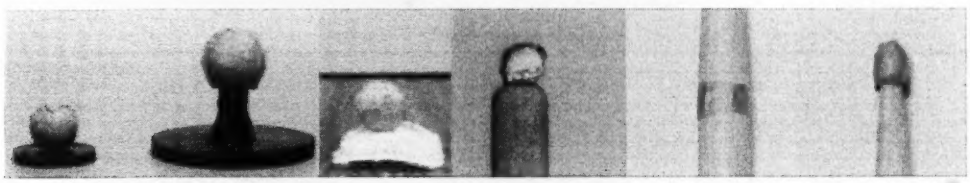


Figure 3.3: Passive retro-reflective markers used with motion capture system. From left to right: markers supplied by Motion Analysis for biomedical and motion picture applications respectively, fabricated thumb marker, and stick butt, shoulder and tip markers respectively.

Analog inputs

The Eagle system also includes a 64 channel, 16 bit analog to digital converter to collect analog data. This module is integrated well into the rest of the system, as it is synchronized with the camera data, and can be controlled from the EVaRT software. Most Gait analysis laboratories are equipped with force plates and an Electromyograph module, both of which interface to the motion capture system through the analog to digital converter module. The sampling rate of the module is in the range of 60 to 5000 Hz, and must be a whole number multiple of the camera capture rate (in order to synchronize the two signals). The input voltage range is ± 50 mV to ± 10 V. The force plate module, as well as additional hardware connected to the analog to digital module were used in this work, and will be discussed in more detail.

EVaRT software

EVaRT software is part of Motion Analysis eagle digital motion capture system. It integrates, and allows control of all hardware components (including camera control and calibration, and analog input control), and provides a large number of motion capture and post-processing tools, and trial archiving functions described below. EVaRT also provides real-time visualization of motion the

capture trial, by displaying all real and virtual marker locations, and subject models.

Tracking and templates

The system can track the trajectory of any number of markers placed in the capture volume. However, this is not of much use if the markers have no meaning or identity. It is therefore necessary to identify each marker to be used in a trial with a unique name, usually related to its position on the body. Together, all the markers used to capture the motion of a subject compose a marker set. In addition to a marker set, markers whose relative distance is more-or-less constant can be connected with linkages. A linkage records the maximum and minimum distance between its two markers during a particular capture trial. Together, a marker set and linkages compose a template of the subject. In fact, a template in necessary for each capture session. A template helps the system track markers in three ways: first, it allows for the automatic recognition of markers when the system first starts tracking. When a marker temporarily disappears (due to occlusion), it allows the system to identify the marker again once it has reappeared. Finally, templates allow the system to distinguish between markers in the capture volume that belong to the marker set, and those that do not. Templates, as well as other configuration information are store in project files. It should be noted that both the configuration and specific location of the markers is important, and will be discussed further in Section 4.3

Post processing tools

Post processing allows for the viewing and editing of motion capture data. All post-processing is done on tracked marker data. Occlusion of markers is inevitable in optical motion capture systems, and is most often caused by a part of the subjects body coming in between the markers and cameras. Occlusion results in two problems: missing data for the occluded markers, and occasionally, when two markers are close together, switched trajectories. EVaRT has tools to deal with both of these problems. Missing data can be recreated by using cubic spline interpolation between visible data points. However, this method only works well for small gaps in data. To correct switched trajectories, EVaRT allows the trajectory of any two markers to be switched within any time range. EVaRT possesses several other post-processing tools, but only these were necessary during post-processing to 'clean up' the motion capture data. During capture trials, switching of marker trajectories occurred occasionally with markers that were very close together (such as in the hand), and had to be manually corrected. Discontinuities in data were infrequent, and small enough that cubic spline interpolation reconstructed the missing data well.

Virtual markers

Virtual markers are tools provided by EVaRT to represent non-existing markers relative to existing ones. Virtual markers can be created to represent non-existing markers in any configuration, but

their most useful and powerful application is in representing the joint centers of a subjects body (such as shoulder, elbow, wrist). To create virtual markers, two or three real markers are used to define two or three coordinate axes. The virtual marker location is then described as a displacement along each of the defined axes. Virtual markers will be discussed in greater detail in Section 4.3.

Outputs

The system can record motion capture data in five formats for archiving, post-processing, and evaluation. The raw video data from each of the eight cameras can be saved. The tracked 3-D trajectories of each marker (real and virtual) can be saved in binary or ASCII formats. If a regular video camera is available, color video data can be recorded in the AVI format. Analog data from external sources captured with the analog to digital converter can be saved in binary or ASCII formats. All formats except for color video data were used in capture trial. Since all analysis of motion capture data was done using Matlab, ASCII file formats instead of binary were used in this work.

3.3.2 System performance

The ability of a motion capture system to precisely calculate the position of a marker is affected by the quality of the calibration. A properly calibrated system is essential for reliable data.

Calibration

Calibration of this system is performed using a two step approach: first a calibration square (L shaped object) with markers at each end and at at the vertex is used to establish a global coordinate system. Next a calibration wand (wand with markers placed at fixed, known distances from each other) is used to improve the calibration iteratively and correct for lens distortion. Most systems report the results of a calibration in some way. This system reports calibration results by the mean value and standard deviation of the calibration wand length.

Accuracy and precision

The performance of a motion capture system can be described by its accuracy and precision. Kadaba and Stine [39] define accuracy as the ability to estimate relative distance and angle between sets of markers, and precision as the ability to reproduce relative distance or angle measurements during repeated, sequential data acquisition [39].

In [72], Richards compared the performance of several motion capture systems including a six camera motion analysis HiRes system. The apparatus used to evaluate four performance characteristic of the systems is shown in Fig. 3.4. The device consists of two markers rigidly fixed 500 mm apart on an aluminum bar, that rotated at approximately 60 rpm. Three additional markers are fixed on a plate mounted at right angles to the aluminum bar; the three markers define the vertices's of a triangle. Another marker is mounted on the base of the device on a 3 cm rigid post. One final

marker is located on the bottom of the rotating aluminum rod, at approximately the same height as the marker fixed to the base of the apparatus. The position of this marker along the length of the aluminum rod can be adjusted. The following performance parameters were calculated for each

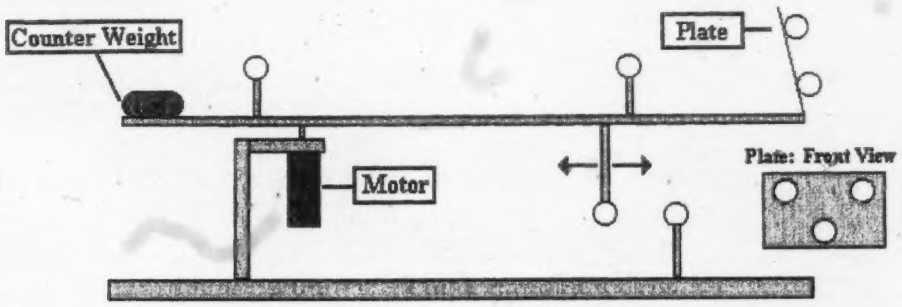


Figure 3.4: Device used by Richards to test the accuracy and precision of a motion capture systems.

system: (1) its ability to measure the distance between two constantly visible markers rotating in the volume, (2) its ability to measure motion associated with a static marker, (3) its ability to reconstruct position-time histories of markers that were visible to alternating sets of two or three cameras, (4) and its ability to measure the motion of a marker that moved in close proximity to a second marker. Of these four performance characteristics, only the first is discussed here in order to allow comparison between the results of Richards work and other evaluations of this system, in which only results for this performance parameter are available.

Richards expressed the systems accuracy as the difference between the actual distance between the two markers, and the mean measured distance over all frames of data. Precision was expressed as the mean squared error (MSE) of the difference between the mean measured distance and the measured distance in each frame of the data. Trials were collected at 60 Hz for 10 s.

A similar study by Kadaba et al. [39] evaluated the performance characteristics of an eight camera motion analysis Eagle system using same apparatus and methods, but only presented results for performance characteristic 1. Accuracy was expressed in the same manner as in Richards [72], but precision was expressed using standard deviation instead of RMS of the distance between two markers. Trials were collected for 5 s, but the capture rate was not mentioned.

The performance characteristic described above was evaluated for the motion capture system used in this work (eight camera Eagle unit) using similar methods. Instead of the apparatus used in [72] and [39], a rigid aluminum bar with two one inch markers fixed 500 mm apart was used (in fact, this was the systems calibration wand). The wand was waved and spun in random orientations and directions within the capture volume. Trials were collected at 240 Hz for 10 s. Accuracy and precision were described as in [39]. The results of the three analysis are summarized in Table 3.1.

Thus, for a given marker location, the reported position is within 1 mm of the actual position,

	Actual mm	Mean Measured mm	Accuracy mm	Precision
Richards et al.	500	497.96	2.04	0.59 RMS
Kadaba et al.	500	499.65	0.35	0.04 SD
This work	500	500.07	0.07	0.34 SD

Table 3.1: Results of accuracy and precision studies of Motion Analysis system. In all cases the precision is greater than 1 mm. In two of three cases, including this work, the accuracy is also greater than 1 mm.

and varies less than 1 mm.

3.3.3 Additional Hardware

In total, three additional pieces of hardware were used in motion capture trials; two were used directly, and one indirectly.

Force plate

Two Kistler force plates are installed in the motion capture laboratory. The plates exhibit good linearity, hysteresis, sensitivity, and a high natural frequency. These properties are summarized in Table 3.2 as per the product data-sheet. The Z axis force-output voltage relationship of the force plates as supplied by the laboratory technician was 1.45 mV per 100 g.

Range	FX, FY	-10 to 10 kN
	FY	-10 to 20 kN
Linearity		$<\pm$ 0.5% FSO
Hysteresis		<0.5% FSO
Natural Frequency		1000 Hz
Sensitivity		<250 mN

Table 3.2: Kistler force plate properties as per manufacturers specifications.

One component of the drumming task to be measured was the impact force when the drum stick struck the playing surface. The properties of the force plate were well suited to measure this, and the initial aim was to use the force plate to measure the impact force. However, having the playing surface at floor level places the drummer in an unnatural and uncomfortable position. To create a normal drumming position, an adjustable stand was placed on the force plate to elevate the playing surface to comfortable level. However, the inertia of the stand reduced the effective natural frequency and sensitivity of the force plate enough that the individual drumming impact forces would not register on the force plate. Consequently, the force plate was abandoned as a method of measuring drumming impact force, and a different method described below was used.

Drum force pad

The drumming surface used in this work was a commercially available drumming practice pad that consists of a block of wood about one inch in thickness, with a section of rubber glued to either

side. The rubber is designed to simulate the rebound provided by a real drum, and thus provided the drummer a familiar surface on which to play. The setup is shown in Fig. 3.5. To measure impact forces, three force sensors were attached to the bottom surface of the drum pad. The readings of each sensor were combined to give drumming impact force. A similar setup has been used by Hajian et al. [29] in a study involving drumming technique, except that a single piezoelectric sensor was used. Flexiforce thin-film force sensitive resistors were used as the force sensors. The resistance of these

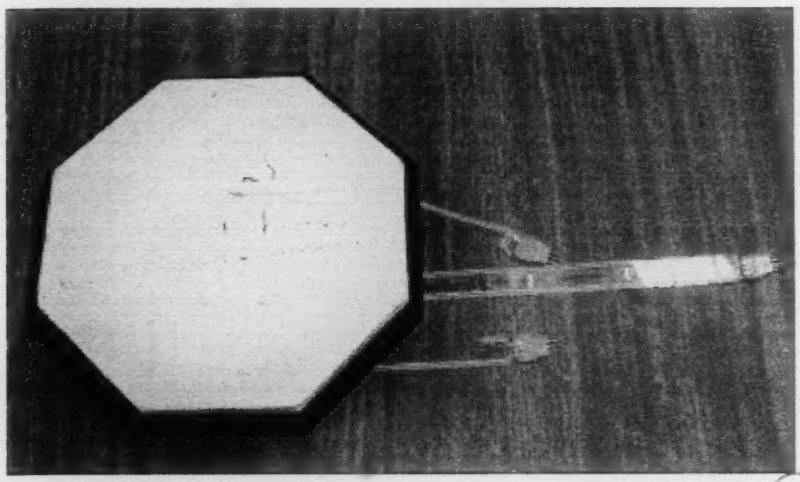


Figure 3.5: Drumming force pad, shown with force sensors attached.

sensor decreases in proportion to the force applied. Their properties are summarized in Table 3.3.

Range	0 to 25 N
Linearity	$<\pm 5\%$, 50% FSO
Hysteresis	$<4.5\%$ FSO
Rise time	$<20\ \mu$sec

Table 3.3: Flexiforce force sensitive resistor properties as per manufacturers specifications.

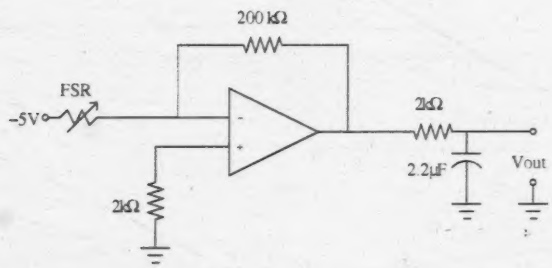


Figure 3.6: Force sensor conditioning circuit: current to voltage converter, and low pass filter.

To convert the sensor resistance to a voltage that could be read by an analog to digital converter, the

circuit shown in Fig. 3.6 was used. Also present is an RC low pass filter with a cutoff of 30 Hz to reduce noise. The output of the three resistor conditioning circuits were connected to the analog to digital module of the motion capture system.

Calibrating the force-output voltage relationship of each sensors was accomplished in the following manner: first the sensor was placed on the calibrated force plate. A constantly increasing force was applied to each sensor until it saturated, the force was then decreased until no force was being applied. The output of both the force plate and sensor were recorded. Since the force-output relationship of the force plate was known, the force plate data could be used to establish force-output voltage relationship of the sensor. Figure 3.7 shows the calibration curve for one of the sensors. Although fairly linear, the relationship is somewhat noisy. The calibration results were stored as a lookup table for each sensor. To convert voltage readings to corresponding forces, the output voltage from each sensor was assigned the force value for that voltage reading in its calibration lookup table.

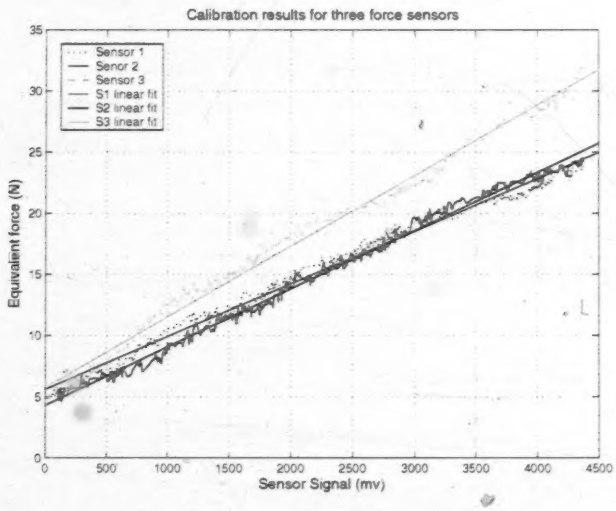


Figure 3.7: Input voltage vs force calibration curve for three force sensors.

3.4 Capturing human motion

3.4.1 Marker set

When creating a markers set for motion capture, several requirements were taken into account: first, the parts of the body for which motion was to be captured, and the type of motion to be captured were considered. At least three markers are required on a given segment to determine all six of its DOF's (X,Y,Z position and orientation). Another consideration when deciding on a marker set were the requirements of any virtual markers that would be defined. Recall that three markers with constant relative positions are necessary to define virtual markers. This implies that all three markers

are rigidly attached to the same segment or body part. Real markers must be attached to a segment to create the trajectories needed for the joint center estimation related to the segment. In addition, a markers' visibility must be considered, as it must be visible by at least two cameras at any time. A final important consideration when selecting a marker position is the amount of skin deformation that will occur at the point of attachment. The significance of these requirements are explained in section 4.

A standard marker-set for bio-mechanical analysis is shown in Fig. 3.8. This marker-set was used as a starting point for marker-set selection, and was altered to satisfy the above constraints and criteria: two markers were used for the wrist (one on either side) instead of one (on top), as they were necessary to define the wrist joint center, and because marker noise due to skin motion was less at these points. Two markers were used on the hand (on the pinkie and index knuckles) to obtain a more robust wrist joint center estimation than a single marker would yield. A marker was added to the thumb to define the fulcrum joint center, and to align the stick segment coordinate system mentioned in chapter 4. Finally, three markers were added to the drumstick. The butt and tip markers were necessary to define the fulcrum virtual marker and stick segment, and to create the trajectory for the fulcrum joint center estimation. Though it was not actually used in any way, the stick shoulder marker was introduced to prevent the motion capture system from switching the trajectories of the stick tip and butt markers, thus reducing the amount of post-processing required. The marker names, attachment locations, function, and partitions are summarized in Table 3.4.

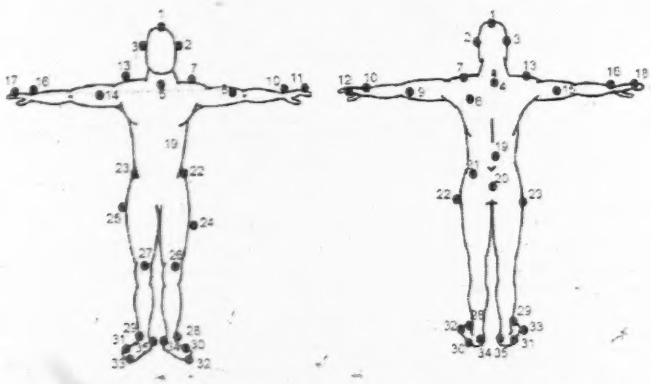


Figure 3.8: Standard full body bio-mechanical marker set recommended by Motion Analysis.

3.4.2 Complete setup

The complete experimental setup included 14 real markers (11 on the subject, three on the drumstick), a three-sensor force pad, eight high speed cameras, and a metronome to maintain a steady drumming frequency. The signal form each of the force pad sensors was run through a low-pass filter with a 30 Hz cutoff frequency, and sampled at 480 Hz. The camera capture rate was 240 Hz.

Name	Attachment Location	Function	Segment
Back	On spine, level with bottom of ribcage (around first or second lumbar vertebrae)	Lower back and neck virtual marker definition	Torso
Neck	On cervical bump (most prominent vertebrae)	Lower back and neck virtual marker definition	Torso
Sternum	Distal end of sternum	Lower back and neck virtual marker, Torso segment definition	Torso
Shoulder	On acromio-clavicular joint	Shoulder joint center virtual marker definition	Collar, Bicep
Bicep	Anterior side of bicep, center	Elbow joint center virtual marker definition	Bicep
Elbow	Poster distal humerus, above humero-ulnar joint	Elbow joint center virtual marker definition	Bicep, Forearm
Radial wrist	Radial styloid process	Wrist joint center virtual marker definition Trajectory for elbow joint center estimation	Forearm
Ulnar wrist	Ulnar styloid process	Wrist joint center virtual marker definition Trajectory for elbow joint center estimation	Forearm
Index	Index finger, distal metacarpal, side of hand	Trajectory for wrist joint center estimation	Hand
Pinkie	Pinker finger, distal metacarpal, side of hand	Trajectory for wrist joint center estimation	Hand
Thumb	On right thumb nail	Fulcrum joint center virtual marker definition Stick segment definition	Stick
Stick tip	Retro-reflective tape around tip	Fulcrum joint center virtual marker definition Stick segment definition	Stick
Stick butt	9 mm strip of tape around shoulder of drumstick		
Stick shoulder	Rigidly fixed to butt end of stick	Fulcrum joint center virtual marker definition	Stick

Table 3.4: Location and configuration of real markers, and their relation to the body model.

Real markers include a back, sternum, shoulder, bicep, elbow, and thumb marker, two wrist and two hand markers (one on either side of each), and three drumstick markers (tip, shoulder, and butt). Figure 3.9 shows the subject with all markers attached, ready for a motion capture session. Not visible are the back and thumb markers.

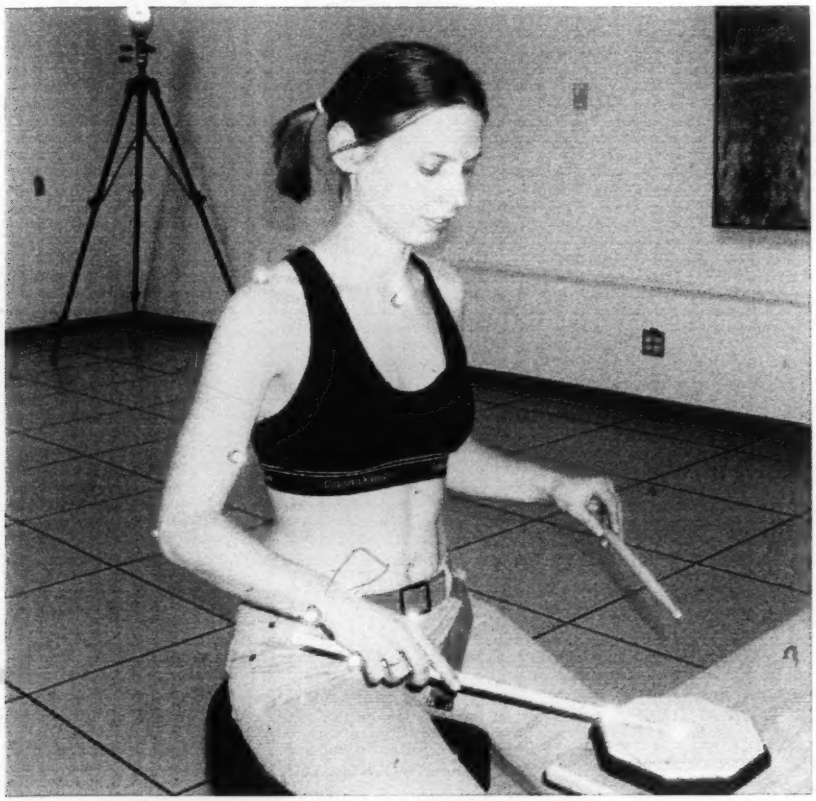


Figure 3.9: Subject with passive retro-reflective markers attached, and ready for motion capture trials. Not visible are the back, neck, and thumb markers.

3.4.3 Capture trials

Three volunteer subjects were used for the capture trials for this work: two were professional percussionists and teachers in the city of Edmonton, one was a semi-professional hobbyist drummer, and all had at lest 15 years experience playing drums. The three particular subjects were chosen based on their ability to perform at least two of the three drumming tasks (described below) effectively over a wide range of playing speeds and dynamics, and perhaps more importantly because they all employed similar drumming techniques, thus facilitating the analysis of the motion capture data for trends[1].

[1] Several widely different methods, or, techniques exist for grasping drumsticks and producing the required strokes. The aim of this component was not to compare different techniques, but rather to obtain quantitative data of a drumming task from which general trends can be observed. Therefore subjects with similar technique were desired.

Discussion with private instructors, professional performers, and university and college percussion instructors highlighted several important skills, or, strokes necessary in drumming. Initially, three of these strokes were selected drumming tasks, as they required fundamentally different motions to produced the stroke. These skills are described below.

Single stroke/roll

The Single stroke is a core skill for all types of drumming, and consists of a simple right/left alternating sequence, producing one note per stroke.

Double stroke/roll

This is a more advanced skill, requiring two notes to be produce per stroke. After the initial stroke, the hand is held momentarily while the stick is allowed to bounce once, and then retracted, producing a right-right/left-left sequence.

Moller stroke/roll

This is also an advanced skill, and is a referred to as a dual-motion stroke. The forearm produces the first note, allowing the stick to rebound freely. As the forearm retracts, the wrist produces a second note. The notes alternate between the right and left hand, producing a right/left sequence.

The capture trials consisted of each stroke, or skill, being performed at three playing speeds (or frequencies) and three playing volumes (or dynamic levels), resulting in a total of nine trials for each stroke. All trials had a duration of 30 seconds. All strokes were performed at single hand frequencies of 2 Hz and 4 Hz, and at the maximum frequency the subjects were able to maintain at a steady rate for 30 seconds[2]. Dynamic levels were soft, moderate, and loud, and the actual levels were determined by subject.

The different drumming strokes or skills used in the trials are generally associated with different genres of drumming, and as a result not all subjects were capable of performing all three drumming tasks. Subject one and two performed the single and double stroke, and subject three performed the single and Moller strokes. Table 3.5 summarizes the tasks performed by each subject, as well as the playing speeds.

	Single stroke (bpm:Hz)	Double stroke (bpm:Hz)	Moller stroke (bpm:Hz)
Subject 1	60, 120, 152 : 2, 4, 5.1	120, 160, 200 : 4, 5.3, 6.7	N/A
Subject 2	60, 120, 184 : 2, 4, 6.1	120, 160, 200 : 4, 5.3, 6.7	N/A
Subject 3	60, 120, 176 : 2, 4, 5.9	N/A	60, 120, 192 : 2, 4, 6.4

Table 3.5: Trials stroke type and single hand playing frequencies performed by each subject.

[2]The subjects were directed to play 16^{th} note strokes at quarter note values of 60 and 120 beats per minute as indicated by a metronome, which corresponds to single hand frequencies of 2 Hz and 4 Hz respectively.

3.5 Results

Motion and impact force data was captured successfully for all trials and subjects, with the exception of the first gym motion for subject one, which was found during post-processing to be corrupted because of a loss of camera synchronization. Motion capture session were approximately 1.5 hours in duration for each subject. Figures 3.10 and 3.11 show example stick tip trajectories and impact force for single stroke for subject one. Peaks occur twice as often for impact forces as they do for tip trajectory because both hands were striking the force pad. Some post-processing was required to correct swapped trajectories (primarily between the stick tip and butt markers). Results are not presented for all subjects and trials for brevity, but were similar for all subjects and trials.

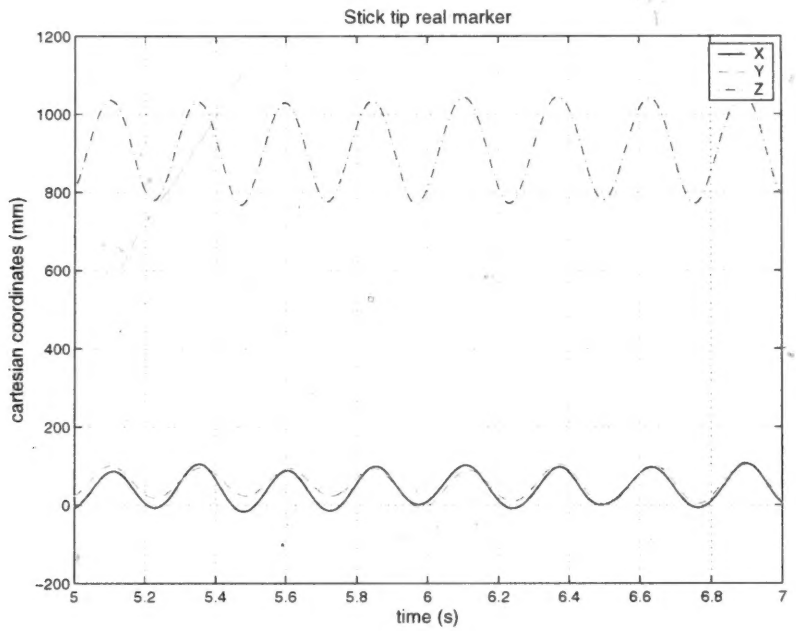


Figure 3.10: Partial time series of stick tip real marker trajectory, subject one, single stroke, 60 bpm, high dynamic level.

3.6 Conclusions

With the exception of the required post-processing (which proved to be time consuming and tedious due to the large number of trials, and their long duration) and occasional loss of camera synchronization, the Motion Analysis system functioned extremely well and with no problems. Calibration was straight forward, and eight cameras made occlusion of markers easily correctable by post processing. Integration of external analog sensors was straight forward, and worked well.

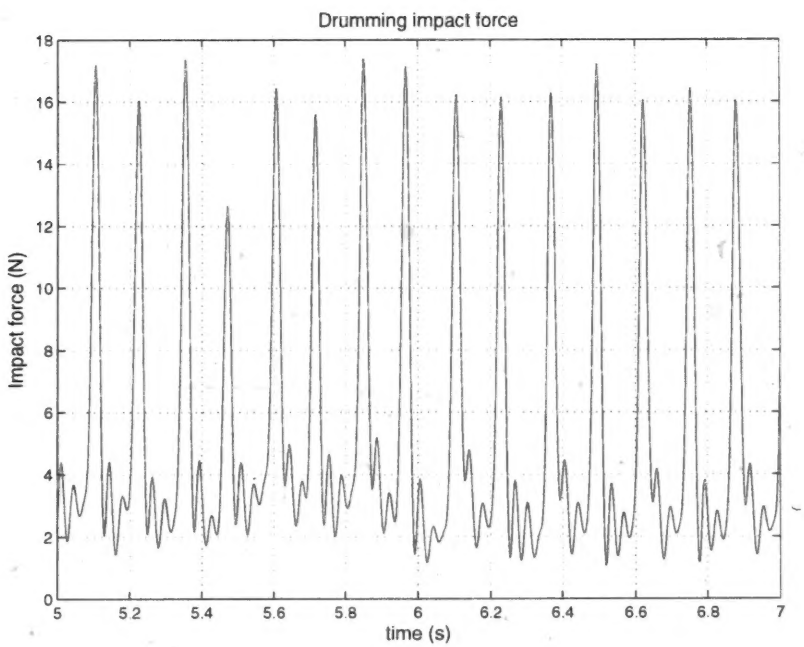


Figure 3.11: Partial time series of drum impact force, subject one, single stroke, 60 bpm, high dynamic level.

Chapter 4

Generating Robot Kinematic Data From Optical Motion Capture Systems

4.1 Introduction

This Chapter will describe the methods and equipment used to obtain kinematic data form optical motion capture systems, that may be used in robot control and imitation learning. An overview of the basic issues to overcome in obtaining kinematic data from motion capture systems will first be outlined, and the methods used in this work will be described in detail. An attempt to evaluate the spatial accuracy of system in reconstructing human motion is then made, and the results compared to human spatial accuracy for the type of repetitive tasks for which motion was to be captured and recreated.

Motion capture has been used in computer graphics applications to animate virtual characters [23][24][16][58], and in the video game and motion picture industries (in movies such as Titanic, Lord of the Rings, Matrix, Final Fantasy) extensively to drive animated characters with excellent results. Motion capture has recently proved a powerful technique for creating realistic looking full body motion for humanoid robotics. It has been used in real time ([74]) and offline to produce kinematic motion for robots [85][43], and it is also being used for imitation learning of humanoid robots [32][9][47][73]. The specific methods used to produce kinematic data depends on the type of motion capture technology used, and are generally only applicable to the motion capture technology they were developed for. All issues mentioned below apply specifically to optical motion capture systems.

Motion of robotics manipulators can be described in terms of manipulator kinematics or dynamics. Kinematics describe motion of an object without regard to the forces that cause the motion. Dynamics describe the forces and moments required for the desired motion to take place. Motion capture systems can only measure the motion of objects, thus only kinematic variable are observable in visual imitation learning [34]. Forward kinematics describe the position of a robotic manipulator

in joint space, that is, in terms of the position of each of its joints. But a set of joint angles alone is meaningless; additional information about a robots configuration is necessary. Specifically, the lengths of the robots links and the organization of the joints must be known. The parameters we wish to obtain from motion capture systems then, are the robots joint angles, segment lengths, and relative joint configuration. Additionally, determining the segment lengths implies the knowledge of the location of the joint centers. In short, we wish to uncover the underlying skeletal arrangement. However, motion capture systems do not measure any of these parameters directly; they can only report the 3-D coordinates of markers on the surface of a body. The problem then, is to to determine the joint centers, joint rotations, segment lengths, and joint arrangement of a subject from the 3-D coordinates of a group of points on the surface of a subject, and this will be the focus of this section. Some issues of generating kinematic data from motion capture system are described first. Next, the methods used in this work are described, and implementation details discussed. Finally, results are presented for actual motion capture trials.

4.2 Issues in generating kinematic data from optical motion capture system

Generating humanoid motion from motion capture data involves relating human joint motion to 3-D marker motion, and though several methods exist to accomplish this, they all have some common features: It is well known that human joints are not perfect rotatory joints, and thus do not have a fixed center of rotation [44]. The typical approach in the bio-mechanics community is to estimate the instantaneous center of rotation. However, as the purpose of the kinematic data is to drive either a robot with fixed rotational joints or a computer graphics character, joints are modeled as ideal rotatory units with fixed centers.

Underlying all the methods discussed here is a body model of the object whose motion is to be captured. This model is in the form of a set of articulations connected by rigid segments. Articulations represent joints, and segments represent rigid body parts, usually underlined by a bone. The body model topology is defined by a hierarchy tree and may be of any shape which allows for tree-like representation of the connected articulations. Most often the motion capture subjects are human. In this case, there is no agreement among different works in the level of complexity required by the model in terms of number of segments and DOF's to properly capture human motion, particularly with regard to the shoulder and spine region.

All work done in this area acknowledges the presence of inherent noise in the marker data due to skin deformation caused by either movement of massive muscles (such as thigh and bicep) or by stretching and accumulating of skin near joints. Some works attempt to address this noise by modeling marker movement relative to the subjects bones, or attaching less importance to noisy markers. Also, most methods rely on minimization of an objective function (usually an error function) to determine the optimal value for some parameters. To determine the parameters in question,

most methods require the subject to perform a motion or set of motions that exercise all the relevant DOF's. This motion, usually called the gym motion, differs slightly among the methods in that some require subjects to exercise one DOF at a time, while others are less strict in the specific motions required. Finally, all methods have some degree of error associated with them due to inherent assumptions or simplifications.

Marker partitioning

To describe the body model, and estimate joint centers and rotations, all methods require knowledge of which markers belong to, or, are attached to which segment. Although this can easily and quickly be done manually and only need be done once for each model, some authors suggest an automated tool. Silaghi et al. [82] for example automatically partition markers by selecting the biggest sets of markers in which the difference between the longest and shortest distance (distance variations) between all pairs of markers over all frames are under a user defined threshold. A specified number of partitions is enforced to promote proper partitioning. The partitions are then assigned to the corresponding segments manually.

Body model

As mentioned previously, the skeleton of a body model is defined as a set of articulations connected by rigid segments. The model topology is defined by a hierarchy tree. The articulations represent the performers joint centers, and the rigid segments his or her bones. A body model is described by the translations and rotations that must take place to get from an articulation to its parent articulation in the hierarchy. That is, it describes the relative orientation and position of the axis of rotation of a DOF with respect to the previous DOF along the tree. In addition to specifying the number of segments and DOF's , several other conventions which impact the overall formulation of the the body model must be chosen. These include coordinate systems (both world and local-body), mathematical conventions for describing the structure and movement of the articulations and segments, and modeling of the marker locations. Each of these components are discussed below.

Number of segments and DOF's: Selecting the appropriate number of segments and DOF's of a body model is a trade off between accurately capturing detailed motion, and maintaining the cost and complexity of the target robot to a minimum. It is desirable to keep the number of segments and DOF's to the minimum necessary to capture the required level of detail in the movement of a particular subject during a specific activity. Several software packages include their own body model, and several modes have been proposed by the bio-mechanics community. Though no formal model exists, a 3-segment 7 DOF model is generally accepted and often used [70] for the arm.

Coordinate systems: In general, two coordinate systems are used when generating kinematic data from motion capture systems. The world or global coordinate system is defined by the calibration sequence of the motion capture system, which generally specifies the Z axis as vertical. The world

coordinate system is necessary to determine the location of markers and thus of the performer in 3-D space. The local or segment-local coordinate system is local to the rigid segment to which it is attached. Furthermore, one local coordinate system exist for each rigid segment.

Usually, segment-local coordinate systems have their origin located at the proximal joint center of the segment, with a primary or bone axis connecting the proximal and distal joint centers. The remaining two axes are defined by one additional non-collinear point, itself defined by either one or a group of markers on that segment. The additional non-collinear point is necessary to orient the axis of rotation (and rotation direction) of a particular DOF. For example, for a bicep segment the origin would be located at the shoulder joint center, and the primary axis would connect the shoulder and elbow joint centers. The non-collinear point can be located in any number of different places to obtain the desired axis of rotation orientation and direction. Rotation and displacement of a segment is then described relative to its parent segment. Additionally, there usually exists a root segment in the hierarchy whose rotation and translation is describes with respect to the world coordinate system.

Mathematical conventions for describing structure and motion: As mentioned previously, a body model is described by the translations and rotations that must take place to get from an articulation to its parent articulation in the hierarchy. One method of describing the relationship from one articulation to the next which borrows from conventional robotics notation, is to represent it by a homogeneous transform matrix of the form.

$$T = \left[\begin{array}{ccc|c} R_{11} & R_{12} & R_{12} & L_x \\ R_{21} & R_{22} & R_{23} & L_y \\ R_{31} & R_{32} & R_{33} & L_z \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

where R_{ii} are coefficients for rotation about the X, Y, or Z axes, and L_i are the displacements along each of the axes. The 3-D position and orientation (in world coordinate system) of any articulation along the tree is obtained by recursively multiplying all the transform matrices that correspond to the preceding DOF's in the tree:

$$X_w = \prod_t T_t(S) \cdot X_L$$

where $X_{l,w} = [x, y, z]_T$ are the segment-local and world coordinates of the articulation, and T_t is the homogeneous transformation matrix, which depends on the state vector S containing the actual rotations about each DOF from the root joint's first to the reference joint's last DOF. This yields the state of the subject P at any instant.

Marker modeling

As mentioned previously, generating humanoid motion from motion capture data involves relating human joint motion to 3-D marker motion. Once the local coordinate systems have been defined for each segment, the location of the markers can be modeled by their location in the corresponding local coordinate system. The markers are usually modeled as being rigidly attached to a segment, that is, their coordinates (in the segment-local coordinate system) are fixed. Relating human motion to

marker motion can then be accomplished by first calculating the world 3-D position and orientation of the corresponding segment as mentioned previously, then transforming the coordinates of the markers form the segment-local to the world coordinate system.

The gym motion: exercising all degrees of freedom

The so called gym motion is a motion the subject must perform which exercises all the DOF's of all the joints for which kinematic data is to be generated. Generally, the motion can be an arbitrary flailing of limb in any direction, though in some cases the subject is required to exercise one DOF at a time. This motion is necessary to create the trajectory needed to estimate the joint center of rotation using the method described below.

Joint center estimation

One method of joint center estimation relies on the hypothesis of a constant distance between markers and joints. Referring to Fig. 4.2, consider parent p_seg and adjacent a_seg segments defined by centers AC and CB respectively. If m is a marker on a_seg, when a_seg moves with respect to p_seg, the trajectory of m will describe a sphere centered at C. To estimate C, the center of rotation of m, the trajectory of m is first reduced so its motion is described relative to p_seg. The center of rotation is then estimated by minimizing an objective function that calculates the difference between the distance from the marker to the center of rotation, and the length of the radius of rotation. The joint center is then determined by taking the center of mass of the rotation center of each marker. This method has also been used with different attachment weights placed on markers to reduce the effect of noisy markers [82]. One considerable disadvantage of this method of joint center estimation is that it requires the subject to perform the gym motion. If motion of a young child or subjects other than humans is being captured, it may be difficult or impossible to obtain the subject to perform this motion, and thus a good joint center estimate will be difficult to obtain.

Another method instead estimates joint centers as offsets from selected skin sites of surface markers, expressed as a fraction of the distance between individual markers as determined from measurements of a single normal male adult [71]. The magnitude of the specific offsets were determined by direct measurement of adult and pediatric skeletons, and by anatomic data available in the literature based on cadavers. Though this method does not require for the subject to perform the gym motion, the joint center location is sensitive to the actual location of the markers on the subject's skin. This method of joint estimation also suffers from a relatively poor ability to estimate the joint center of children or non-human subjects, as joint center location data is only available for adult human specimens.

Segment length estimation

Having estimated the joint locations, the length of the segments can then be determined in a number of ways. One simple approach is to compute the length of a segment as the mean distance between

the two corresponding joints over all frames of data. Another approach is to compute the length that minimizes the MSE of deviations between the segment length and the marker-to-marker distance. Yet another approach is to calculate the segment length that minimizes the difference between the joints of a subject model and the calculated joint centers over all frames of data [82].

Joint angle estimation

Using an LSQ framework and the marker model described previously, one can estimate a subjects joint angles in each frame by minimizing the distance between measured markers positions, and modeled marker positions generated by the recovered joint angles. This approach is used in [22][85][82] with some variations. Ude et al. [85] use the recovered state (joint angles) of the previous frame as an initialization for the LSQ minimization of the current one. Silaghi et al. [82] instead use the estimated joint center locations in each frame as an initialization for the LSQ minimization. A disadvantage of this method is that is does not run in real time, and is thus strictly a post processing operation. Also, none of the works using this method present quantitative result of its accuracy.

A more direct direct method of estimating joint rotations is as follows. First, local coordinate system are defined with their origin at the proximal joint center of a segment, and the principal axis connecting the proximal and distal joint centers. The remaining two axes are aligned with the axis of rotation of the DOF's of a particular articulation using a third non-collinear marker. The rotation of a segment with respect to its parent segment is then calculated by finding the rotation matrix relating the two coordinate systems. Although this method is straightforward, a combination of the choice of coordinate system, rotation order, and actual observed motion can produce widely different results, though with proper consideration, good results can be obtained [71].

Marker noise

One inherent characteristic of motion capture systems is noisy marker data due to skin deformation where the markers are attached. The skin deformation is caused by either movement of massive muscles (such as the thigh and bicep), or by stretching and accumulating of skin near joints. With respect to estimating joint centers, some works [82] distinguish between skin deformation that changes a markers' distance from or along the bone, and deformation that only causes markers to move around the bone at a fixed distance. Markers that are known to exhibit the former type of movement are classified as noisy markers. This problem is addressed by assigning all markers an attachment weight, with noisy markers being assigned a lower weight than non-noisy ones in order to reduce the effects of noise. The attachment weights are user defined, and can be adjusted to improve results, though the improvement has not been quantified.

4.3 Methods

Following is a description of the methods used in this work to obtain kinematic data from motion capture data.

Marker set and partitioning

Partitioning of markers into their respective segments was straight forward, and since it only needs to be done once, was done done manually. Note that markers located near or on joints belonged to two segments. The marker names, attachment locations, function, and partitions are summarized in Table 3.4.

Body model

Number of segments and DOF's:
As mentioned previously, selecting the number of segments and DOF's of a subject model is a trade off between the accuracy of the captured motion, and complexity of the model and target robot. The number of segments are in general closely related to the underlying skeletal structure. Knowledge of the subject's kinematics can be used in some cases to keep the number of DOF's of the model to a minimum. For this work, only the right side of the upper body was modeled, using a total of seven segments and 15 DOF's.

The torso was modeled as a single segment connecting the lower back joint center and the shoulder girdle joint center, with 3 DOF's (flexion/extension, abduction/adduction, and rotation) at the lower back joint center. The underlying skeletal component of this segment was the spine, which is not rigid. This segment then did not take into account well motions that curved spine, and some loss of accuracy was incurred due to the simplification. Motion of the torso was measured with respect to the global calibration coordinate system. The sternum marker was used to indicate the direction of the torso.

The collar bone was modeled as a segment connecting the shoulder girdle joint center and the right shoulder joint center. This part of the body is only capable of motion relative to the torso in the flexion/extension and abduction/adduction directions, and not rotation. This segment was therefore initially modeled as having two DOF's. However, a third DOF was added to accommodate three resulting joint angles due to choice of coordinate systems and rotation orders. Motion of the collar segment was measured with respect to the torso segment, with the lower back marker indicating the direction.

The bicep, or upper arm segment was modeled as a segment connecting the right shoulder joint center and the elbow joint center, with three DOF's at the shoulder (flexion/extension, abduction/adduction, and medial/lateral rotation). Motion of this segment was measured with respect to the collar segment. Since the elbow joint is not capable of movement in the abduction/adduction direction, the plane defined by the bicep and forearm segments can be used to define the rotational

direction of the bicep segment. Thus the wrist marker was used as the direction marker for this segment.

Together, the collar and shoulder segments and respective joint centers are a overly simplified representation of more complex body kinematics, and thus reduce the accuracy or capturing the motion.

The forearm is only capable of motion about the elbow in the flexion/extension and pronation/supination directions, and not in the abduction/adduction direction. This body section therefore also only has two DOF's. The forearm, or lower arm segment was modeled as two segments with one DOF each. The upper forearm section connected the right elbow joint center and the midpoint of elbow and wrist joint center, and measured flexion/extension about the elbow joint center. Motion of this segment was measured relative to the bicep segment, with the shoulder joint center indicating its orientation. The lower forearm segment connected the forearm midpoint and wrist joint center, and measured pronation/supination of the forearm. Motion was measured relative to the upper forearm segment, with the outside wrist marker indicating the direction.

The hand segment is only capable of motion about the wrist in the flexion/extension and abduction/adduction directions, and not in the pronation/supination direction. This segment therefore only has two DOF's. The hand segment was modeled as the segment connecting the right wrist and fulcrum joint centers, with the two DOF's at the wrist. Motion was measured relative to the lower forearm segment, with the outside wrist marker indicating the direction.

The stick segment is capable of three DOF's. However, twist about the sticks axis is inconsequential as it is symmetric about this axis. Thus this segment was modeled as having two DOF's (flexion/extension and abduction/adduction). These segments thus would only require two markers. The drumstick was then modeled as the segment connecting the fulcrum joint center and stick tip, with the two DOF's at the fulcrum joint center. Motion was measured relative to the hand segment, with the thumb marker indicating the direction.

Coordinate systems:

Two coordinate systems were used for this work: A global coordinate system, and a segment local coordinate system. The global coordinate system was defined by the calibration sequence of the motion capture system, which defined the Z axis as the vertical, and the X and Y axis by a special calibration square placed in the capture volume.

Segment-local coordinate systems were defined with the origin located at the proximal joint center of the segment, and with the primary or bone axis connecting the proximal and distal joint centers. The remaining two axes were defined by one additional non-collinear point. The additional non-collinear point was necessary to orient the axes of the coordinate system with the desired axes of rotation (and rotation direction) of a particular DOF. Each of the coordinate axes were assigned to the axes of rotation that corresponded to flexion/extension, abduction/adduction, and pronation/supination of a segment. Additionally, the primary or bone axis is also the axis of rotation

in the pronation/supination direction. Even though this method of describing segment-local coordinate systems is often used for analysis of human kinematics in the bio-mechanical community, there is no consensus on the actual assignment of the coordinate axes (X-Y-Z) to the axes of rotation, nor in the order in which the rotations are calculated [71][90].

For this work three coordinate systems were tested: X-Y-Z, X-Z-Y, and Z-Y-X axes corresponding to flexion, abduction, and supination respectively. Furthermore, X-Y-Z and Z-Y-X rotation orders were considered. One inherent disadvantage of using Euler angles to describe rotation of objects is the presence of gimbal lock [1]. All the considered methods exhibited gimbal lock, though the X-Y-Z axes assignment with the X-Y-Z rotation order exhibited much less than the other methods for the motions carried out by the human subject. For this reason, the X-Y-Z axes assignment corresponding to flexion, abduction, and supination, and X-Y-Z rotation order were used.

The axis vectors describing the segment-local coordinate system were calculated in the following manner. If m_{root} and m_{long} are the proximal and distal joint center markers respectively, and m_{plane} is the additional non-collinear marker, where $m = [X, Y, Z]$ are the coordinates of m in 3-D space, define the two normalized vectors $\vec{rl}$ and $\vec{lp}$ given by

$$\vec{rl} = \frac{m_{long} - m_{root}}{|m_{long} - m_{root}|}$$

$$\vec{rp} = \frac{m_{plane} - m_{root}}{|m_{plane} - m_{root}|}.$$

The $\vec{rl}$ vector is assigned to be the Z axis.

$$Z = \vec{rl}.$$

The Y axis vector is obtained by taking the cross product of Z and $\vec{rp}$

$$Y = Z \times \vec{rp}.$$

Finally, the X axis vector is

$$X = Y \times Z.$$

The coordinate axes are then

$$C = \begin{bmatrix} X_x & Y_x & Z_x \\ X_y & Y_y & Z_y \\ X_z & Y_z & Z_z \end{bmatrix}$$

where the X, Y, and Z axis correspond to flexion/extension, abduction/adduction, and pronation/supination of a segment respectively.

With the exception of the drumstick tip marker, distal and proximal markers were always joint center estimations in the form of virtual markers. Both real and virtual markers were used for the third, non-collinear marker. In some instances, this marker did not align the calculated coordinate

[1] When a rotation of more than $\pm 90°$ occurs about the Y axis, the X and Z axes align and cause a singularity to appear in the rotation matrix. The resulting rotation angles extracted from a rotation matrix with a singularity will not reflect the actual rotations, and will be incorrect

axes with the desired flexion and abduction axis of rotation. The offset was generally $\pm 90°$ or $180°$, and was easily corrected by rotating the coordinate system about the segment bone axis the necessary amount.

Mathematical conventions for describing structure and motion: As mentioned previously, a body model is described by the translations and rotations that must take place to get from an articulation to its parent articulation in the hierarchy. In this work, homogeneous transform matrices were used to describe this relationship.

$$T = \left[\begin{array}{ccc|c} R_{11} & R_{12} & R_{12} & L_x \\ R_{21} & R_{22} & R_{23} & L_y \\ R_{31} & R_{32} & R_{33} & L_z \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

The 3-D position and orientation (in the world coordinate system) of any articulation along the tree were obtained by recursively multiplying the transform matrices that correspond to each of the preceding DOF's in the tree:

$$jc_w = \prod_i T_i(\theta_{xyz}, L) \cdot jc_l$$

where jc_l and jc_w are the local and world coordinates of the i^{th} segment at angle θ and of length l.

With the exception of the first segment in the hierarchy, the homogeneous transform matrices contain translations along the bone axis only which reflects the length of that segment. The first segment in the hierarchy contains translations in all directions, and reflects the displacement of its proximal marker from the origin of the global coordinate system. Table 4.1 summarizes the body model definition.

The gym motion

The gym motion is required to create the marker trajectories necessary for joint center estimation, and other model parameter calibration. For each joint center to be estimated, the subject must exercise all DOF's for the calibration procedure to return valid answer. Two different gym motion were used in this work. One emphasized motion about one DOF at a time, while the other was more flexible. The sequence of the first gym motion consisted of bicep flexion and abduction (approximately $90°$), elbow flexion, wrist flexion, and finally fulcrum flexion. The second one was as follows: For the shoulder joint, the arm was to be moved in a circular manner at the side and from of the body, and tracing circles of different sizes. For the elbow joint, the forearm was to be flexed and extended through the full range of motion. For the wrist joint, the hand was to be moved in circular manner. Finally, for the fulcrum, using the left hand the stick was to be moved in a circular manner. During motion of each segment, the performer was indicated to maintain the parent segment at steady as possible by bracing it with the left hand.

Name	Markers			Parent	DOF's			Z axis	
	root	long	plane	segment	X axis	Y axis	Z axis	Offsets	Motion about joint
Torso	Lower back JC	Shoulder girdle JC	Sternum	Global	Flexion	Abduction	Rotation	-90°	Waist
Collar	Shoulder girdle JC	Shoulder JC	Lower back JC	Torso	Flexion	Abduction	Rotation	0°	Shoulder girdle
Bicep	Shoulder JC	Elbow JC	Wrist JC	Collar	Flexion	Abduction	Pronation	-90°	Shoulder
Upper Forearm	Elbow JC	Forearm midpoint	Shoulder JC	Bicep	Flexion	N/A	N/A	-90°	Elbow
Lower Forearm	Forearm midpoint	Wrist JC	Wrist outside	Upper Forearm	N/A	N/A	Pronation	180°	Elbow
Hand	Wrist JC	Fulcrum JC	Wrist outside	Lower Forearm	Flexion	Abduction	N/A	180°	Wrist
Stick	Fulcrum JC	Stick tip	Thumb	Hand	Flexion	Abduction	N/A	180°	Fulcrum

Table 4.1: Body model segment definitions.

Virtual markers

Virtual markers are a tool provided by EVaRT to represent non-existing markers relative to existing ones. Virtual markers can be created to represent non-existing markers in any configuration, but their most useful and powerful application is in representing joint centers. Four types of virtual markers are available: two-marker value, two-marker ratio, three-marker value, and three-marker ratio. To define a 3-marker virtual marker, three existing non-collinear markers are required and are called the root, long and plane markers respectively. The three markers are used to create a set of coordinate systems with its origin at the root marker. The axis are created in the following manner. If m_{root}, m_{long}, and m_{plane} are the root, long, and plane markers respectively where $m = [x, y, z]$ are the coordinates of m in 3-D space, define the two normalized vectors $\vec{rl}$ and $\vec{rp}$ given by

$$\vec{rl} = \frac{m_{long} - m_{root}}{|m_{long} - m_{root}|}$$

$$\vec{rp} = \frac{m_{plane} - m_{root}}{|m_{plane} - m_{root}|}.$$

The normalized Y axis of the new coordinate system is simply

$$\vec{y} = \vec{rl}.$$

The $\vec{z}$ axis vector is obtained by taking the cross product of $\vec{rp}$ and $\vec{y}$

$$\vec{z} = \vec{rp} \times \vec{y}$$

Finally, the X axis vector is

$$\vec{x} = \vec{y} \times \vec{z}.$$

The location of the virtual marker is then specified as a scalar offset o, from the root marker in each of the three directions:

$$vm = m_{root} + \vec{x} \times o_x + \vec{y} \times o_y + \vec{z} \times o_z.$$

The offsets can be specified in millimeters, or as a ratio of the unnormalized magnitude of $\vec{rl}$. Furthermore, previously defined virtual markers can be the root, long or plane marker of subsequent virtual markers. Similarly, three-marker virtual markers are calculated as a displacement (in millimeters or as a ratio), along $\vec{rl}$ from m_{root}.

EVaRT provides a graphical interface for defining virtual markers. Figure 4.1 shows the interface for the virtual marker definition of the shoulder joint center. Clearly visible are the Root, long and plane markers, and the offset in millimeters along each axis. In total, six virtual markers were included in the body model. The definitions for each are shown in Table 4.2. All virtual markers we used to represent joint centers. The values for the actual offsets are discussed below.

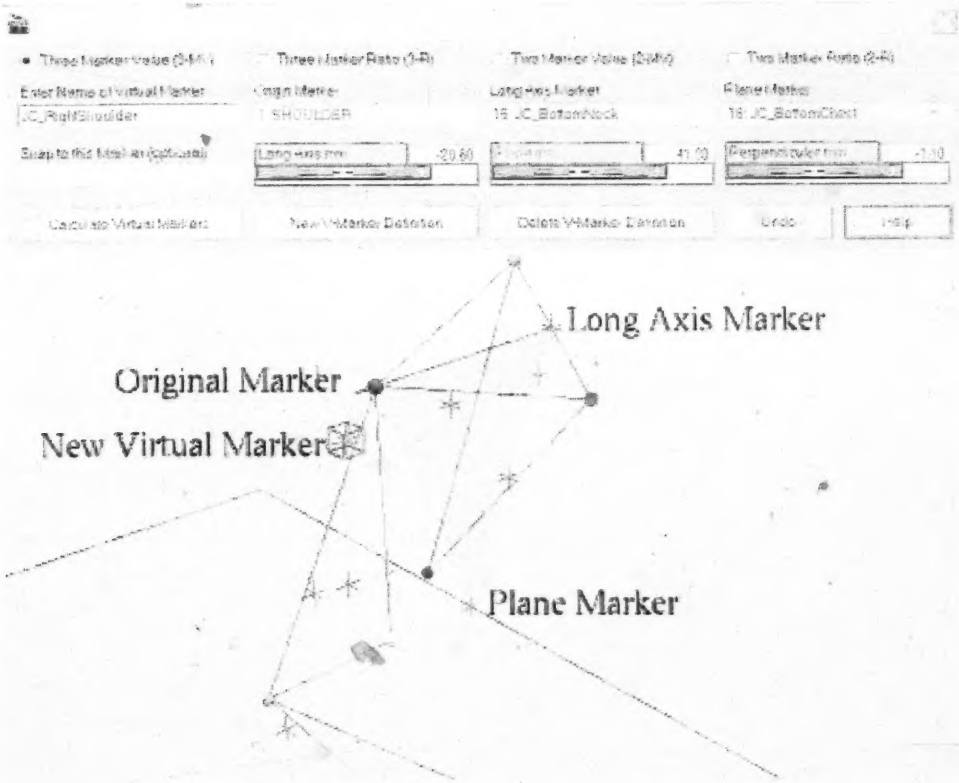


Figure 4.1: EVaRT virtual marker visualization and definition tool.

Virtual marker name	Type	Root marker	Long marker	Plane marker
Hip joint center	3-MV	Mid back	Rear	Sternum
Shoulder girdle center	3-MV	Neck rear	Sternum	Mid back
Shoulder joint center	3-MV	Shoulder	Neck joint center	Mid back
Elbow joint center	3-MV	Elbow	Shoulder joint center	Bicep
Wrist joint center	3-MV	Wrist outside	Wrist inside	Elbow joint center
Fulcrum joint center	3-MV	Thumb	Stick tip	Stick butt

Table 4.2: Body model joint center virtual marker definitions.

Joint center estimation

To estimate the center of each of the subject's joints, the method of Silaghi et al. [82] was used. Referring to Fig. 4.2, marker m is rigidly attached to a_seg, and moves with respect to p_seg with a spherical trajectory centered at C. If the estimated center of rotation of marker m is given by cr, where $m, cr = [x, y, z]$, and the constant radius of rotation is r, then the distance d from cr to the actual center of rotation C is given by

$$d(r, cr_x, cr_y, cr_z) = \sqrt{(cr_x - m_x)^2 + (cr_y - m_y)^2 + (cr_z - m_z)^2} - r. \tag{4.1}$$

To estimate the center of rotation, the parameters cr and r must be selected that minimize 4.1 over a specified trajectory

$$argmin_{r,cr_x,cr_y,cr_z} \sum_{trajectory} (d(r, cr_x, cr_y, cr_z))^2. \tag{4.2}$$

The joint center estimation is then obtained by taking the center of mass of the center of rotation of all the markers on a_seg.

$$jc = \frac{\sum_i^n cr_i}{n}. \tag{4.3}$$

This method was extended by representing the estimated joint centers with virtual markers in the following manner: Recall that the coordinates of a virtual marker are given by the three reference markers, and scalar offsets o in each axis direction

$$vm = m_{root} + \vec{x} \times o_x + \vec{y} \times o_y + \vec{z} \times o_z. \tag{4.4}$$

The distance e from a joint center jc to its representative virtual marker vm is given by

$$e(o_x, o_y, o_z) = \sqrt{(vm_x - jc_x)^2 + (vm_y - jc_y)^2 + (vm_z - jc_z)^2}. \tag{4.5}$$

To accurately represent the joint center with the virtual markers, the offsets o that minimize 4.5 must be selected

$$argmin_{o_x,o_y,o_z} \sum_{trajectory} (e(o_x, o_y, o_z))^2. \tag{4.6}$$

This procedure of minimization to estimate the joint centers and represent them using virtual markers must be done once per capture session for every subject. Once this procedure has been completed, and the virtual marker offsets determined for each joint, the joint center coordinates can be easily obtained in subsequent capture trials of a capture session using equation 4.4.

A special case occurs when a marker trajectory describes a circle instead of a sphere. This occurs when the a_seg has only one DOF, such as the elbow, and the joint center can lie anywhere along the axis of rotation. In this case, the trajectory of the markers on that segment are first projected onto the plane that best fits the trajectory. The plane is found as follows: The distance D from a point $p = [x, y, z]$ to a plane described by the parameters a, b, c, d is given by

$$D = \frac{|ax + by + cz + d|}{\sqrt{a^2 + b^2 + c^2}}. \tag{4.7}$$

Finding the parameters a, b, c, d that minimize 4.7 over the trajectory of p yields the plane that best contains that trajectory

$$argmin_{a,b,c,d} \sum_{trajectory} (D(a,b,c,d))^2. \tag{4.8}$$

Once the best fitting plane has been found, the trajectory can be projected onto it, and the joint center calculated as above.

To transform the trajectory of a marker so it is with respect to its parent segment, the coordinates of the root marker of the corresponding virtual marker to be calculated were simply subtracted from the coordinates of the markers associated with that segment.

As this work was primarily concerned with arm movements, an accurate estimate of the joint centers of the shoulder girdle and pelvis were not necessary. Instead of this method then, values suggested by Motion Analysis in example models for bio-mechanics applications and the EVaRT manual were used.

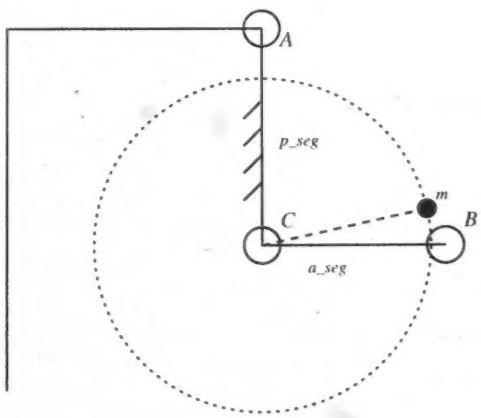


Figure 4.2: Trajectory of marker m rigidly attached to segment a_seg, with respect to parent segment p_seg.

Segment length estimation

Having estimated the joint center locations, the length of each segments was taken as the length that minimized the MSE between the segment distal joint center recovered from the model, and the measured distal joint center as described by the corresponding virtual marker. The recovered distal joint center position first was calculated by recursively multiplying the homogeneous transform matrices of each segment from the root segment to the current segment

$$jc_r = \prod_i T_i(\theta_{xyz}, L) \cdot jc_l \tag{4.9}$$

where jc_r are the recovered world coordinates of the i^{th} segment distal joint center at angle θ and with length l. The distance between the recovered and measured joint centers is given by

$$\Delta jc = \sqrt{(jc_{r_x} - jc_{m_x})^2 + (jc_{r_y} - jc_{m_y})^2 + (jc_{r_z} - jc_{m_z})^2}. \tag{4.10}$$

where jc_m is the measured distal joint center. Minimizing equation 4.10 over all frames of data yielded the length of best fit for that particular segment.

$$argmin_L \sum_{trajectory} (\Delta jc_{trajectory}(L))^2. \tag{4.11}$$

To decrease the optimization time, the mean distance between the two joint centers over all frames of data was used as the initial estimate of the segment length for the minimization task. Segment lengths were calculated recursively outward from the root segment in the chain to the last one. The length of each segment was calculated individually, with all preceding segment lengths kept constant (at their estimated lengths). In this manner the recovered and measured proximal joint center for the root segment coincided.

Joint angle estimation

To estimate joint angles, the method of Rab et al. [71] was used. With a segments coordinate system C calculated as described previously, the rotation from C_{parent} to C_{seg} about each axis was given by

$$R_{rot} = R_{parent} \times R_{seg}^{-1}, \tag{4.12}$$

where R_{rot} is a 3×3 rotation matrix containing the coefficients of rotation about each of the axes. The rotation angles about each axis were then extracted from R_{rot} in the following manner: Rotation about all three axes in the order X-Y-Z yields the coefficient matrix

$$\begin{bmatrix} c(\theta_z)c(\theta_y) & -s(\theta_z)c(\theta_y) & s(\theta_y) \\ c(\theta_z)s(\theta_y)s(\theta_x) + s(\theta_z)c(\theta_x) & -s(\theta_z)s(\theta_y)s(\theta_x) + c(\theta_z)c(\theta_x) & -c(\theta_y)s(\theta_x) \\ -c(\theta_z)s(\theta_y)c(\theta_x) + s(\theta_z)s(\theta_x) & s(\theta_z)s(\theta_y)c(\theta_x) + c(\theta_z)s(\theta_x) & c(\theta_y)c(\theta_x) \end{bmatrix} \tag{4.13}$$

where cos and sin have been abbreviated to c and s. The angles of rotation about each axis are given by

$$\theta_y = asin(R_{13})$$

$$\theta_x = -atan(\frac{R_{2,3}}{R_{3,3}})$$

$$\theta_z = -atan(\frac{R_{1,2}}{R_{1,1}})$$

where $R_{i,j}$ are the rotation coefficient in the $(i,j)^{th}$ entry of R, and are shown below.

$$R_{1,3} = s(\theta_y)$$

$$R_{2,3} = -cos(\theta_y)sin(\theta_x)$$

$$R_{3,3} = cos(\theta_y)cos(\theta_x)$$

$$R_{1,2} = -sin(\theta_z)cos(\theta_y)$$

$$R_{1,1} = cos(\theta_z)cos(\theta_y)$$

Marker noise

To minimize the inaccuracies associated with marker noise due to skin deformation relative to the skeleton, all markers (with the exception of the bicep marker) were placed over prominent bony land marks, where the subcutaneous tissue is thin and relatively fixed to the underlying skeleton. The bony landmarks were also easily identifiable, thus facilitating the consistent placing of markers on the subject. The bicep marker deviated due to flexing of bicep muscle, which deforms maximally in a direction radial to the bone. This marker was used exclusively as the plane marker for the virtual marker definition of the elbow joint center. Deflection of this marker in a direction radial to the bone has a minimal effect on the resulting orientation of the virtual marker plane, and thus only slightly affects the final value of the virtual marker.

4.4 Implementation

EVaRT was used for live motion capture of subjects, and for limited post processing. Post processing consisted of correcting swapped marker trajectories during tracking, and filling in missing data using cubic spline interpolation. The tracked marker trajectories were saved in ASCII format, and were later read into Matlab for further processing.

All other calculations were implemented in Matlab, including virtual marker, segment coordinate system and joint angle calculations, and joint center, and segment length estimation. The Matlab function lsqnonlin was used (with boundaries, and large scale algorithm using a trust region to determine the direction and size of optimization step) for all optimization tasks. Prior to all calculations, tracked marker data was passed through a low-pass digital filter with a cutoff frequency of 30 Hz. For ease of use, a graphical user interface shown in Fig. 4.3 was created from which to initiate all operations and calculations. Model parameters were saved in a text file and Matlab .mat file. The animated robot is shown in Fig. 4.4 performing a recovered gym motion. Trajectories for segment length estimation were taken as the entire gym motion capture trial. For joint center estimation, the trajectory was taken as the portion of data where motion about the joint of interest was exercised, and was chosen manually. For visual display of the recovered model and forward kinematics calculations, the Matlab robotics toolbox [18] was used.

4.5 Results

To analyze the performance of the proposed methods, data from three subjects of different sizes performing two instances of the gym motion were used.

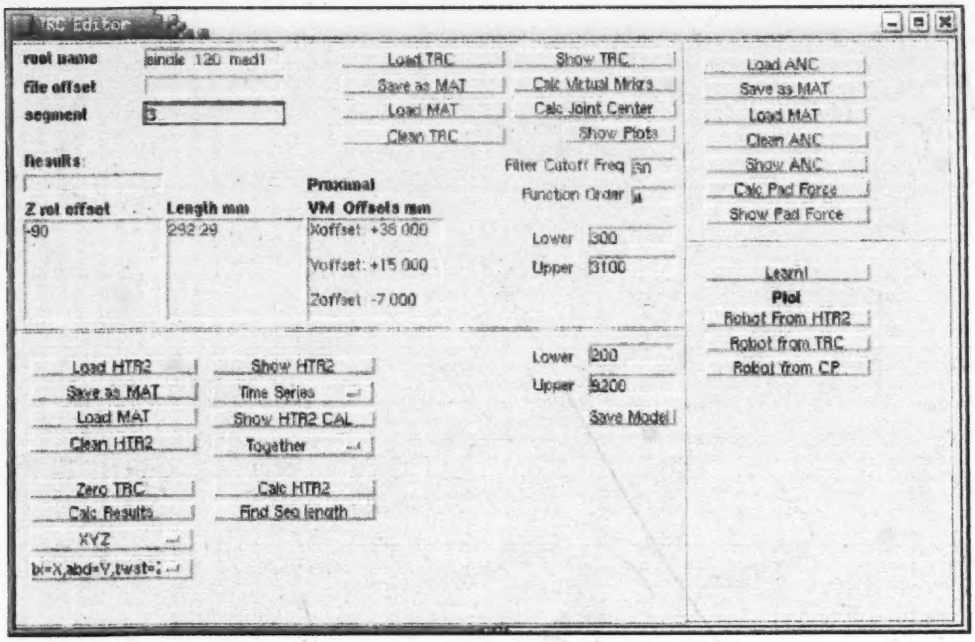


Figure 4.3: Graphical user interface used to initiate all tasks.

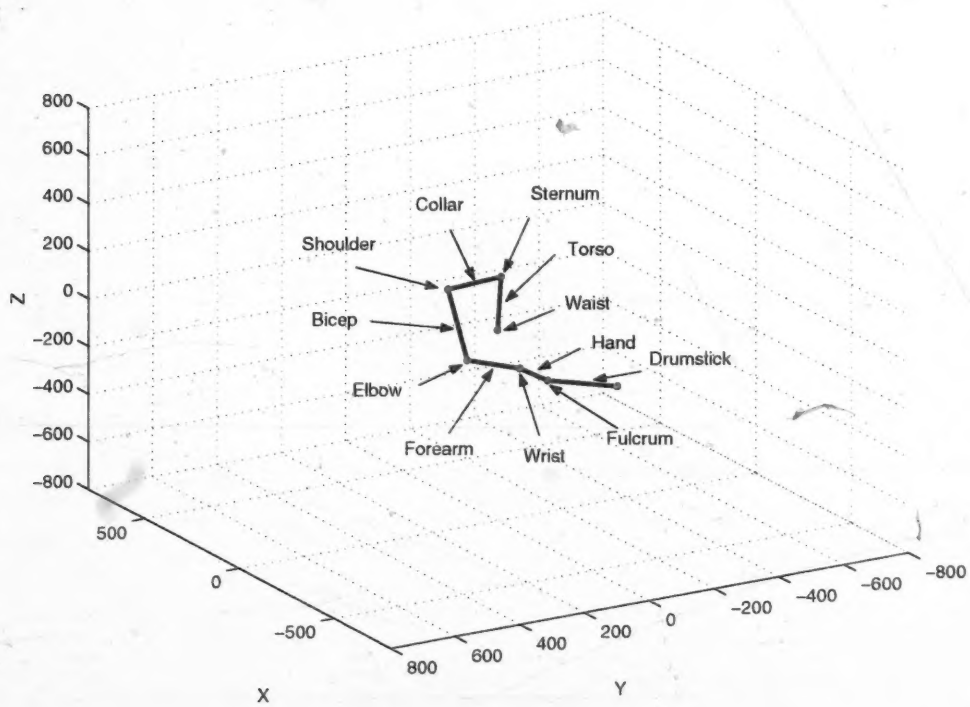


Figure 4.4: Animated model of robot created using Matlab robotics toolbox.

Two specific performance aspects of the methods are analyzed: The accuracy of the joint center estimation procedure, and its sensitivity to the data are first examined. The overall accuracy of the methods in recreating human motion is then examined. Next, sources of error are considered and their effects are presented. Finally, the results are put into perspective by comparing them to human performance in repetitive or rhythmic tasks.

Joint center estimation

The resulting virtual markers offsets for the shoulder, elbow, wrist, and fulcrum joint centers are shown in Table 4.3. The offsets are presented for each of the two gym motion trials as absolute offsets in millimeters, and for comparison with expected joint centers obtained by [71] and from observations of [42], as a percentage of the associated segment length.

Offsets for the shoulder joint center virtual marker were consistent throughout all subjects and trials, as well as consistent with the expected real-world location of the joint center. The largest offset was in the vertical direction, directly down from the shoulder marker, with only slight offsets in the lateral and anterior directions.

Offsets for the elbow joint center virtual marker were consistent across trials, but showed some variability across subjects, particularly in the Z offset. With straight arm beside subject and pointing directly downward, and the elbow pointing back, the Y and X offsets place the joint center directly below the elbow marker anywhere from -18 to -4 mm, and forward anywhere from 20 to 32 mm respectively. Both of these offsets are more-or-less consistent with the expected joint center location. The Z offset however, places the joint center laterally from the elbow marker anywhere from -8 to 21 mm, not at all as expected. Recall that the elbow is capable of motion in the flexion, but not in the abduction direction. Consequently, the joint center also has a reduced DOF, and can thus lie anywhere on the axis of rotation in the flexion direction. The Z axis offset is then a result of this reduced DOF, and of the method used to handle this special case.

Excluding trial 1 for subject two, offsets for the wrist joint center virtual marker are consistent across subjects and trials. However, the offsets are not entirely consistent with the expected joint center location. With the forearm and hand horizontal, the X and Z offsets place the wrist joint center near the ulnar wrist marker, as expected. The Y offset however, places the wrist joint center only 8 to 14 mm laterally from the wrist marker, not in the midpoint of the two wrist markers as expected. Because of the small offset of the wrist joint center from the ulnar wrist marker, which was also used as the third non-collinear marker to define the hand segment, small variations in the relative position of the wrist ulnar and joint center markers caused a large change in the direction of the axes of the hand segment coordinate system. The resulting joint angles were very noisy, and did not describe motion about this joint very well. As a result, the wrist joint center offsets were not used, and the values suggested by Rab et al. [71] (which place the wrist center at the midpoint of the two wrist markers) were used instead.

Offsets for fulcrum joint center virtual markers were neither consistent throughout trials or subjects, nor with the expected real-world location. In fact, the values were not compatible with the drumstick segment definition, and consequently fulcrum joint center virtual markers offsets were manually adjusted to place the joint center at point passing through drumstick, and close to the thumb marker.

Table 4.4 summarizes the joint center virtual markers offsets values used for the remainder of this work. The shoulder and elbow offsets were taken as the the mean of the two gym motion trials. The wrist offsets were taken from Rab et al., and the fulcrum offsets were selected as described previously.

	mm			% Segment length		
	root	long	plane	root	long	plane
Subject 1, one trial						
Shoulder	41	1	-1	13.8	0.3	-0.3
Elbow	30	-7	21	11.3	-2.6	7.9
Wrist	-7	13	3	-5.4	9.9	2.3
Fulcrum	7	-13	2	2.5	-4.6	0.7
Subject 2, two trials						
Shoulder	32/35	10/0	-6/7	11.9/13	3.7/0	-2.2/2.6
Elbow	24/32	-16/-18	-8/2	10.9/14.6	-7.3/-8.2	3.6/0.9
Wrist	17/-7	2/16	-2/9	11.3/-4.6	1.3/10.6	-1.3/6
Fulcrum	49/4	1/16	-2/-1	16.8/1.4	0.3/5.5	-0.7/-0.3
Subject 3, two trials						
Shoulder	38/38	13/17	-7/-6	13.4/13.4	4.6/6	-2.5/-2.1
Elbow	20/22	-4/-4	18/3	7.9/8.7	-1.6/-1.6	7.1/1.2
Wrist	-8/-1	12/7	1/-1	-6.5/-3.3	9.8/6.5	0.8/-1.6
Fulcrum	-13/-21	-11/11	3/-6	-4.9/-7.9	-4.1/4.1	1.1/-2.2

Table 4.3: Resulting joint center virtual maker offsets. The left columns shows the absolute offsets from the root marker in mm, and the left columns show the offset as a percent of the segment length.

Joint angle estimation

Measured joint angles and segment lengths (joint center to joint center distance) for a trial are shown in Fig. 4.5(a) though (f). The motion for this joint consisted of bicep flexion and abduction, and elbow, wrist, and drumstick flexion with the bicep partially flexed. Movement about the pelvis, or waist, shown in Fig. 4.5(a) consists mostly of rotation in the vertical axis, which reflects the direction the subject was facing during the trial. Forward and side rotation are minimal indicating the subject was sitting more-or-less upright. Rotation of the collar segment about the sternum was expected to be mostly about the Y axis, which would place the collar segment at approximately 90° to the torso, with a minor rotation about the X axis to reflect the slight forward displacement of the shoulders relative to the torso, typical of human posture. Figure 4.5(b) instead shows considerable movement of the collar segment about the sternum about all three axes. This is an example of gimbal lock: rotation about the Y axis is in the area of 90°, and the resulting X and Z angles are not correctly

	mm		
	root	long	plane
Subject 1			
Shoulder	41	1	-1
Elbow	30	-7	21
Wrist	-7	16	9
Fulcrum	32	0	0
Subject 2			
Shoulder	34	5	1
Elbow	28	-17	-3
Wrist	0	midpoint	0
Fulcrum	33	0	0
Subject 3			
Shoulder	38	15	-7
Elbow	21	-4	11
Wrist	0	midpoint	0
Fulcrum	26	0	0

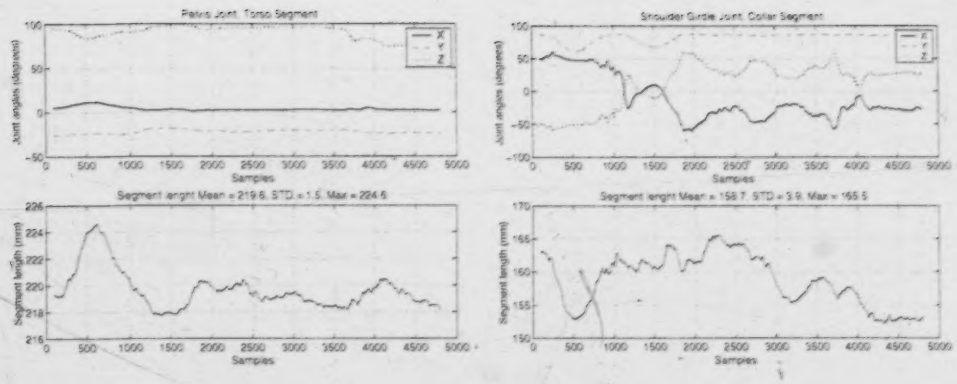
Table 4.4: Joint center virtual maker offsets used in remainder of work.

decomposed. It is because of gimbal lock and the resulting angles that three DOF's instead of two were necessary to model this joint. Though not intuitive to interpret, the recovered angles for the shoulder joint shown in Fig. 4.5(c) are consistent with the expected results. Rotation of the forearm segment about the elbow joint is clearly visible in Fig. 4.5(d), and is as expected. Rotation occurs only about the X (flexion) and Z (forearm twist) axes. Recovered rotation angles for the hand and stick segment (about the wrist and fulcrum joints respectively) are also as expected, and can be seen in Fig. 4.5(e) and (f). Rotation about the Z axis for the stick segment was ignored, as it had no effect.

Accuracy

As a measure of the system's accuracy in reproducing the captured motion, the magnitude of the difference between the reproduced segment distal joint centers (or segment endpoint) and measured joint centers (as represented by their respective virtual markers) over an entire trial was used. This calculation was performed during the segment length estimation task, and is in fact the residual error from the segment length estimation. The resulting residual errors for both trials of all three subjects are presented for each segment distal joint center (or segment endpoint) in Table 4.5, and are expressed as the mean and maximum values, and standard deviation over the entire trial. Also included are the mean, maximum, and standard deviation of the measured segment length (joint center to joint center distance) for the same trial. Additionally, Fig. 4.7(a) through (f) show a typical trajectory and associated residual error for the trial discussed previously. The corresponding measured segment lengths for this trial are shown in Fig. 4.5(a) though (f).

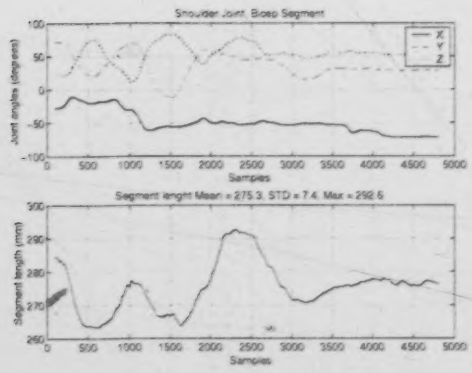
As Table 4.5 shows, the mean end-point errors are consistently in the area of 10 mm, and are always less than 1 % of the total kinematic chain length. The maximum end-point error over all



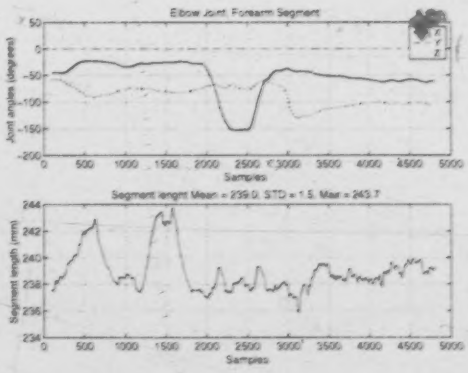
(a) Pelvis joint recovered angles

Shoulder Girdle Joint, Collar Segment
Joint angles (degrees)
Samples
Segment lenght Mean = 158.7, STD = 3.9, Max = 165.5
Segment length (mm)
Samples

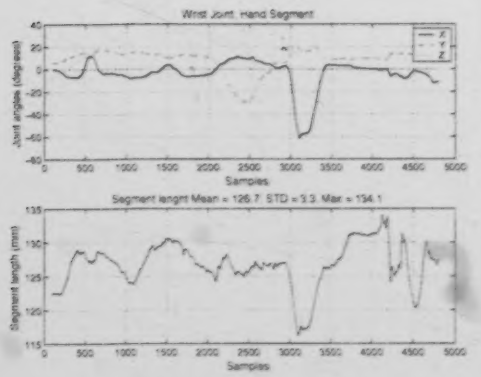
(b) Shoulder girdle joint recovered angles



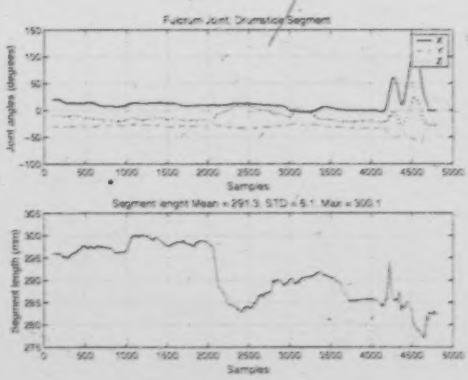
(c) Shoulder joint recovered angles



(d) Elbow joint recovered angles



(e) Wrist joint recovered angles



(f) Fulcrum joint recovered angles

Figure 4.5: Recovered joint angles (top subplot) and segment lengths (bottom subplot) for first gym motion trial for subject two.

subjects and trials is 32.3 mm, about 2.5 % of the kinematic chain length. The largest mean error over all subjects and trials is 13.5 mm, about 1.8 % of the kinematic chain length up to that segment. Segment length estimations are consistent across trials and also consistently deviate more from the trial mean further along in the kinematic chain. In all cases, the maximum deviation of the estimated length from trial mean occurs for the forearm segment. In addition, the bicep segment has the highest standard deviations for residual error and measured length, indicating that the shoulder or elbow real markers, or both, move a considerable amount with respect to the underlying bone. The error between the measured and recreated motion for one frame of data in the trial is visible in Fig. 4.6. A large error is visible at the shoulder joint and stick tip, with smaller errors elsewhere. As mentioned previously, all methods for creating kinematic robot data from motion capture data have inherent errors associated with them. Following is a discussion of the sources of error associated with the used methods, and the relative contribution of each.

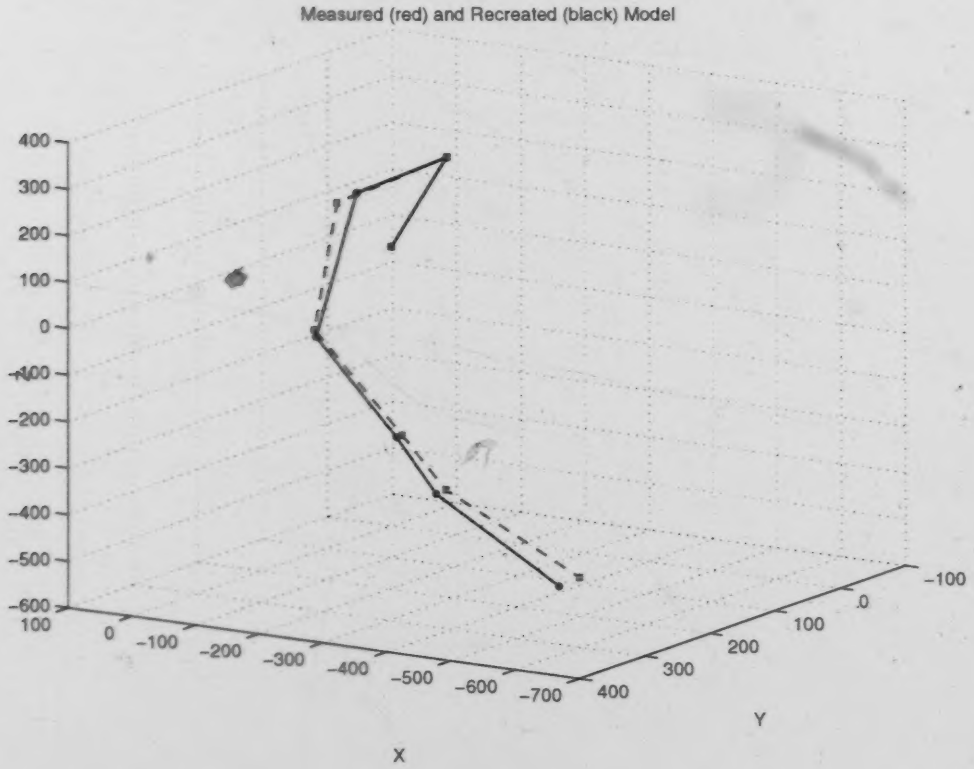


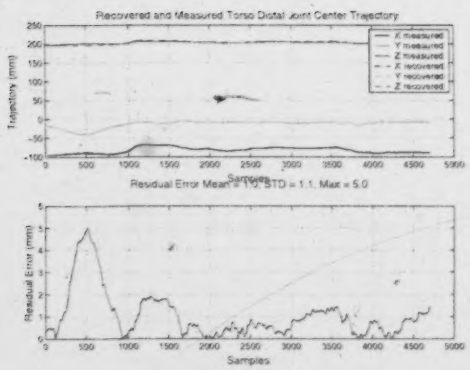
Figure 4.6: Example measured (- ■ -) and recreated (-•-) motion for first gym trial, subject two. Errors are visible at the shoulder, fulcrum, and stick tip.

Sources of error

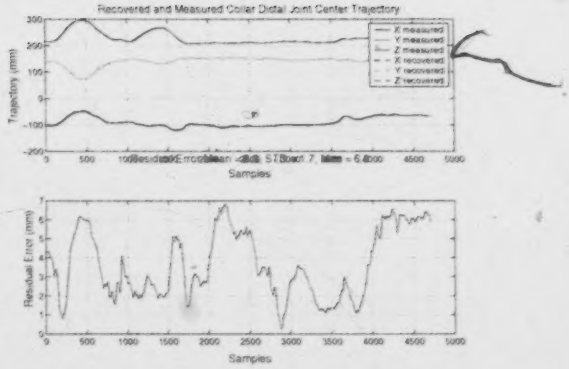
Referring to Fig. 4.7 and 4.5, at each segment the residual error is large when the segment length varies from the trial mean, and the two traces in fact resemble one another for the first few segments.

	Segment					
	Torso	Collar	Bicep	Forearm	Hand	Stick
Subject 1, one trial						
Segment length						
Estimated	265	174	308	256	135	284
Mean	265	174	309	261	136	285
STD	1.0	5.5	12.6	6.4	2.2	2.5
Max	267	185	339	271	143	289
Residual error						
Mean	0.8	4.4	13.5	10.6	10.6	10.5
STD	0.6	3.4	4.7	3.5	3.9	4.1
Max	2.5	13.1	29.4	20.9	21.7	30.7
Subject 2, two trials						
Segment length						
Estimated	220/220	159/160	275/275	236/236	126/123	292/291
Mean	220/220	159/160	275/275	239/240	127/124	291/290
STD	1.5/2.2	3.9/6.8	7.7/7.6	1.5/1.9	3.3/5.7	6.1/8
Max	225/227	166/171	293/294	244/244	134/137	300/313
Residual error						
Mean	1.0/1.6	3.6/6.2	7.6/9.8	7.1/9.5	7.9/9.7	9.5/11.6
STD	1.1/1.6	1.7/3.3	5.5/6.9	4.7/6.2	4.7/6.6	5.4/8.7
Max	5.0/6.7	6.8/11.9	20.8/24.8	19.6/25	18.7/25.1	24.7/32.3
Subject 3, two trials						
Segment length						
Estimated	253/252	178/178	285/285	253/253	131/129	266/265
Mean	253/253	178/178	286/286	260/259	130/128	268/267
STD	0.9/1.5	4.3/4.2	10.9/11.1	2.3/1.8	3.7/3.8	1.6/2.6
Max	256/256	185/185	301/305	279/264	139/135	272/280
Residual error						
Mean	0.7/1.1	3.1/3.0	11.7/12.0	10.0/10.1	9.0/9.6	9.0/9.7
STD	0.6/0.8	3.0/2.9	5.5/5.6	3.8/4.9	3.5/5.1	3.5/5.1
Max	3.3/3.8	16.8/16.1	31.3/32.2	23.5/24.1	23.8/20.3	23.1/20

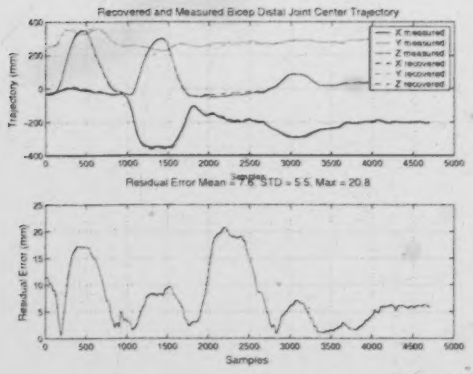
Table 4.5: Resulting segment lengths and joint center residual errors, all units in mm. This table shows, for each segment, the estimated length (as described previously), the mean, STD, and maximum of the measured length (distance between proximal to distal joints). Also shown are the mean, STD, and maximum of the residual error between the reconstructed and measured distal joint centers for each segment. Of most interest, the rightmost column shows the residual error between the reconstructed and measured end effector (drumstick tip) trajectory for each trial.



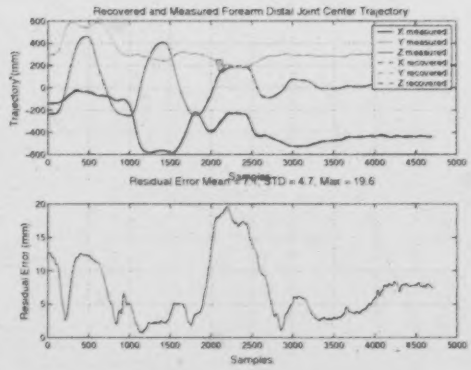
(a) Residual error at torso segment.



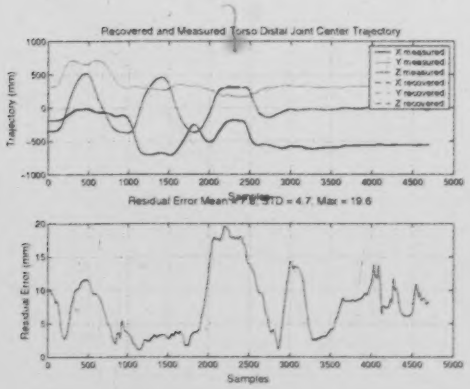
(b) Residual error at collar segment



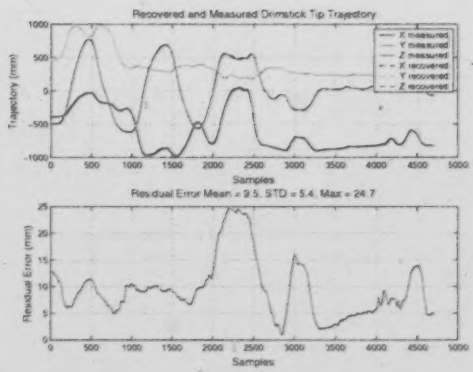
(c) Residual error at bicep segment



(d) Residual error at forearm segment



(e) Residual error at hand segment



(f) Residual error at stick segment

Figure 4.7: Example of recreated distal joint center of each segment for first gym trial of subject 2. The top plot shows measured and reconstructed segment distal joint coordinates. The bottom plot shows reconstruction residual error.

The residual errors at each segment propagates thought the kinematic chain to all segments after it. For example, if at some configuration there is a large residual error at the shoulder joint, a corresponding residual error will be present at the elbow joint, wrist joint, etc. The residual errors at each joint also accumulate along the kinematic chain. For example, comparing the residual error traces and the measured segment length traces for the torso segment, one can see they appear similar, and that the largest residual error occurs when the segment length deviates from its mean value the most.

In the collar segment residual error, one can see effects of the varying collar segment length as well as of the varying torso segment length, and thus the accumulation of errors at each joint. The same is true of the residual error for traces of each segment; the residual error traces show the error at the current segment, and all other segments preceding it. This propagating error is also evident from the increasing mean residual error along the kinematic chain.

Varying segment length, is therefore a major source of error, and has two contributing causes: First, if real markers used to define joint center virtual markers move with respect to the underlying bone during some motion, the respective joint center virtual markers will also move. For example, during large abduction of the bicep, the external shape of the shoulder deforms considerably, and significant skin movement occurs. This will cause the shoulder joint center virtual marker to deflect accordingly, which is not representative of the actual motion of the joint center. Both the shoulder and elbow real markers were observed to deflect in this manner. Also, if the model used to capture a subjects motion is too simple, that is, has too few segments and DOF's to accurately capture the motion taking place, the motion will be described by the model as a change in segment length, not as the actual motion. For example, only one segment was used to represent the torso here. If a subject were to bend forward at the waist and arch their back, the motion would reflect in the model as a large decrease in torso segment length. This was the case for the collar, bicep, hand, and drumstick segments. As Fig. 4.8 shows, the bicep and stick segments showed the greatest variability with standard deviations of 7.4 mm and 6.1 mm respectively, the collar and hand showed moderate variability with standard deviations of 3.9 mm and 3.2 mm respectively, and the torso and forearm segments showed the least variability, each with standard deviations of 1.5 mm. A less marked and visible source of error is noise or movement of real markers used to define a segment's coordinate system. Movement or noise of these markers will manifest itself correspondingly in the resulting joint angle data.

4.5.1 Human accuracy in repetitive motor tasks

There is a well known relationship between accuracy and speed of human motion in directed reaching tasks. This feature has been studied extensively, and for higher movement speeds, where spatial correction through visual feedback is not possible, target accuracy (usually defined as standard deviation of termination position) is well known to decrease as movement speed increases [13][41][60].

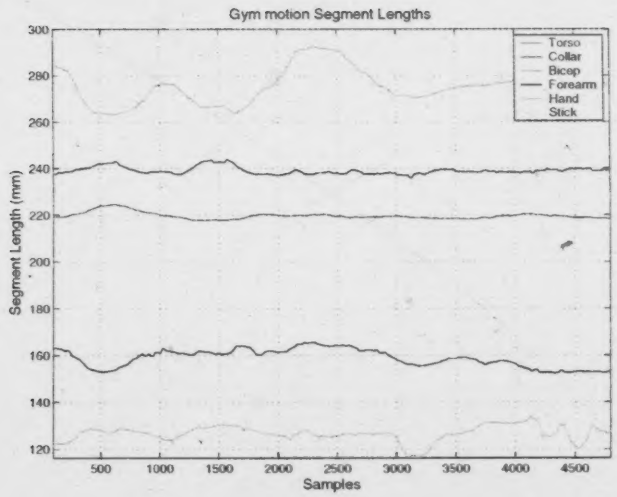


Figure 4.8: Actual length of all segments during first gym motion trial subject two.

This relationship is also true for rhythmic or cyclical task, though to a smaller degree than in discrete movements [28], that is, spatial accuracy varies less in cyclical tasks than in discrete tasks with the same movement time. Additionally, humans exhibit spatial and temporal variability in rhythmic arm movements [36][83]. It is also well know that even short term practice or training of repetitive tasks can increase general performance [21], as well as coordination, accuracy, and execution speed of the task [20][46].

To compare the spatial accuracy of the model with that of the subject's for the drumming task, the magnitude of the standard deviation of the X and Y drumming impact coordinates was used as an metric of accuracy. The impact positions were taken as the X and Y coordinates with the minimum corresponding Z coordinate within a cycle. This index was calculated for all three subjects for one drumming task at several frequencies and playing forces. The resulting standard deviations are shown in Table 4.6, and the impact coordinates for subject two performing the drumming task at 6.13 Hz and at medium force are shown in Fig. 4.9.

As Table 4.6 shows, the minimum standard deviation over all subjects and trials was 2.3 mm. For the single stroke task, the minimum stick tip variability shown by all subjects across all trials was 2.3 mm. Similarly, the average standard deviation over all trials and subjects was 6.2 mm. The mean residual reconstruction error over all trials and subjects was 10 mm, with a 5.4 mm standard deviation. On average then, subjects show 6.2 mm variability in the single stroke task, and error associated with the proposed method is between 4.6 and 15.4 mm.

Drumming frequency	Dynamic level		
	Soft	Medium	Loud
Subject 1			
2 Hz	2.3	7.2	7.4
4 Hz	3.2	5.4	8.5
5.07 Hz	3.6	9.9	10.2
Subject 2			
2 Hz	7.7	7.2	6.6
4 Hz	4.5	5.7	7.2
6.13 Hz	4.9	6.5	9.7
Subject 3			
2 Hz	3.4	5.6	6.3
4 Hz	3.9	5.9	7.4
5.87 Hz	4.6	4.5	8.6

Table 4.6: Magnitude of standard deviation of impact coordinates (X,Y), single-stroke task, all units in mm.

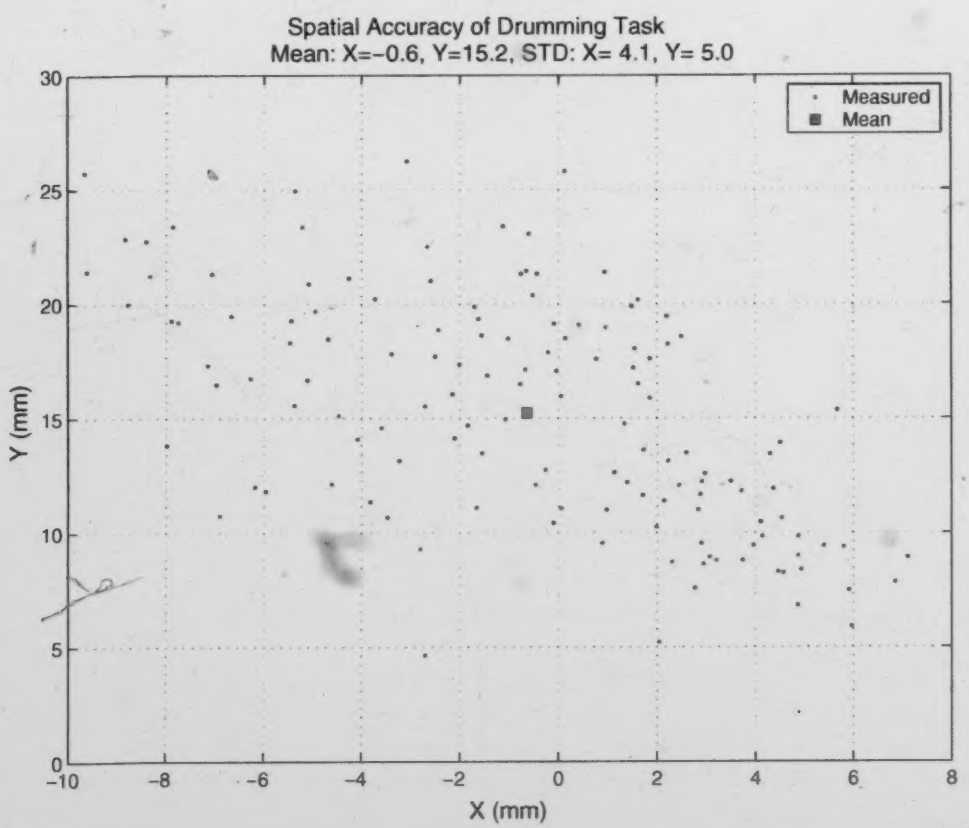


Figure 4.9: Drumming impact coordinates (subject 2, 6.13 Hz, medium dynamic level).

4.6 Conclusions

The method used for estimating joint centers from the motion capture data was not robust as it was very dependent on the actual data used (the actual gym motion performed), and consequently did not yield good results. Also, as mentioned previously, the necessity for a calibration motion such as the the gym motion imposes a considerable constraint on the type of motion capture subjects. The proposed joint center estimate method then, is not sufficiently accurate and robust, and this area should be further pursued. Despite this however, good overall results were still obtained, as the remainder of the methods (joint angle and segment length estimates) used to extract kinematic data from motion capture data were reasonably accurate in reconstructing the measured motion, and were robust to poor joint center estimates. Furthermore, the choice of coordinate systems and rotation order were effective in reducing gimbal lock. When the kinematic data is used as a first step of a goal or model based imitation learning framework, the described methods certainly should provide sufficient accuracy for a good initialization to the optimization process. The described methods also have the capability of dealing with the correspondence problem, as the developed body model can be made to match the body of the target robot, and imposed in the subject by the selection of motion capture marker locations.

Chapter 5

Analysis of Human Motion

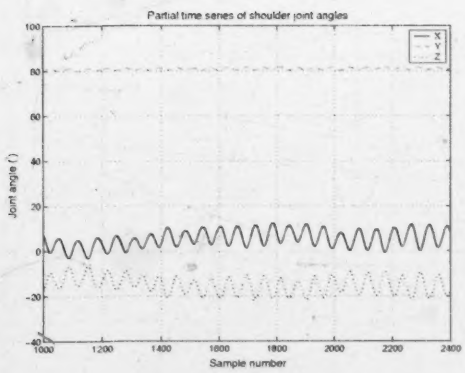
5.1 Introduction

In this chapter, the results of the motion capture trials are analyzed by automatically extracting the inherent characteristics of the oscillations: their frequency, amplitude, phase, and center of oscillation. The reasons for this are two fold: First, the extracted characteristics may be used to determine a priori some of the control policy parameters, and serve as an initialization to imitation learning, thus simplifying the task. Second, the analysis will aim to uncover any qualitative preferred values, visible trends, or relationships in the drumming task across the subjects or trials, which can be subsequently used to modify the learned behaviors for slightly different situation by appropriately adjusting control policy parameters. The chapter will give an overview of the methods used, and summarize the results of the analysis for all subjects for each of the four aspects that were analyzed. The analysis results for individual subjects are not included in this chapter for brevity, but are available in appendix A.

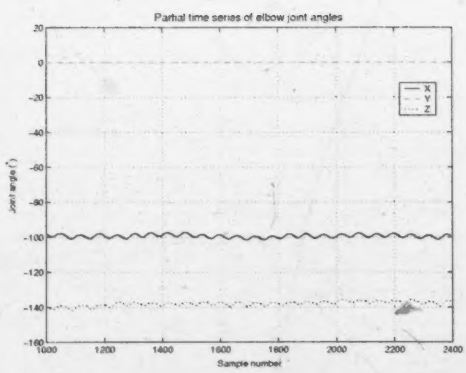
Although motion of all three tasks was captured, and an original goal of this work was to analyze and recreate motion for all three tasks, because of time considerations only one of the three tasks was used in the remainder of this work. The single stroke, as it is both a core drumming skill and a relatively simple one, was deemed an appropriate skill to utilize. Thus, results are presented below for the single stroke drumming task only. Figure 5.1 shows a representative partial time series of the single stroke drumming task played by subject one at 120 bpm (4 Hz) and moderate dynamic level.

5.2 Methods

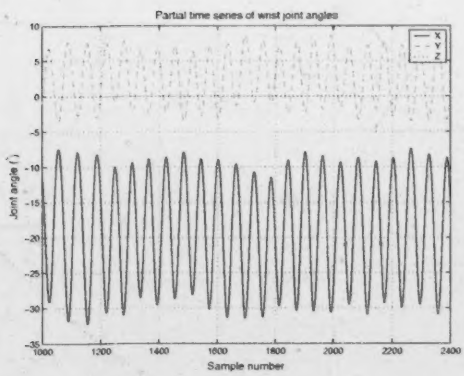
Shoulder, elbow, wrist, and fulcrum recovered joint angles were analyzed along all relevant DOF's. Specifically, the frequency, amplitude, center of oscillation, and relative phase of each were calculated. The frequency, amplitude, and center of oscillation, and phase were calculated by taking a 8192-point Fourier transform of each of the signals. The power spectrum was converted to equivalent amplitude by taking the magnitude of the power spectrum, and scaling it by the length (number of samples) of the signal [50]. Since all drumming was in the frequency range of 2 to 8 Hz, the



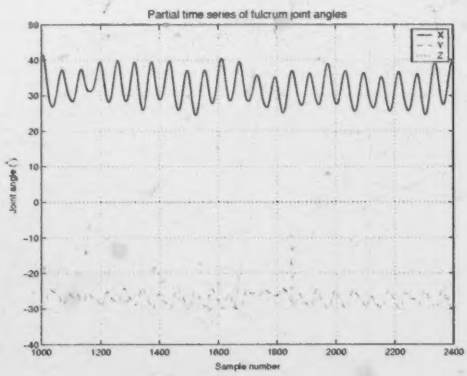
(a) Shoulder angle trajectory.



(b) Elbow angle trajectory.



(c) Wrist angle trajectory.



(d) Fulcrum angle trajectory.

Figure 5.1: Angle trajectories for single stroke drumming task at 4 Hz, subject one. The wrist joint displays the largest oscillations, followed by the fulcrum and shoulder, with the elbow showing little oscillation.

frequency component with peak amplitude within this range was taken as the oscillating frequency. Additionally, only oscillations with an amplitude greater than 1° were considered[1]. The amplitude of oscillation was taken as the peak amplitude within the 2 to 8 Hz range. The center of oscillation was taken as the amplitude of the zero-frequency component in the Fourier transform. The phase of the signal was calculated as the angle between the real and imaginary components of the power spectrum using equation 5.1.

$$\phi = \pi = atan2(imag(P_{max}), real(P_{max})) \tag{5.1}$$

Research on bi-manual coordination in humans and quadruped locomotion in the fields of psychology and neurobiology has highlighted the relative phase of limbs involved in rhythmic motions as an important variable in describing the order of the coordinated oscillations in biological motion [84][27]. Some emergent properties of rhythmic motion in humans and animals have also been brought to light. Specifically, inter-limb oscillatory motion and quadruped locomotion show preferred and in some cases invariant values of relative phase between involved limbs for a variety of tasks in humans and animals [27][69][40]. Furthermore, humans have a strong preference for strictly in-phase (0°) and anti-phase (±180°) relationship for bi-manual rhythmic tasks [84]. The relative phase of the involved segments and components is therefore presented here.

The relative phase was calculated as the difference in the phase values obtained as described above between each segment, and the Z component of the the stick-tip marker trajectory. This trajectory was selected to compare all others to, as it ultimately defines the playing frequency, and because motion will be always present for this component regardless of the type of stroke or dynamic playing level. Figure 5.2 shows a partial time series of the drumstick tip trajectory during a single stroke drumming task played by subject one at 120 bpm (4 Hz) and moderate dynamic level.

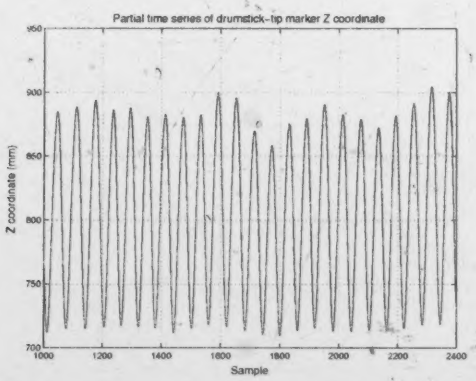


Figure 5.2: Partial time series of drumstick-tip marker Z component trajectory.

[1]Small oscillation were neglected, as they are considered noise. These small motions occur because of the interaction torques across the body. Because of interaction toques in the subjects body, movement of some limbs results in small oscillations in other parts of the body [12]. These small oscillations are considered noise, as learning is to take place only on voluntary movements.

5.3 Results

Following, the results of center, frequency and amplitude of oscillation, and relative phase analysis of the four involved segments and all relevant DOF's are presented. Results of each category of analysis were compared across all three subjects for each joint. Specifically, the comparison attempted to highlight any obvious trends or preferred values in each category with respect to task frequency or dynamic level. For brevity, only the results of the comparisons are presented here. The actual results for all three subjects are included in Appendix A.

Center of oscillation

Shoulder: The center of oscillation for the X component showed some variability, but no apparent trend was visible. Values were more-or-less constant at -6°, -6°, and -43° for subjects one, two, and three respectively. Results for the Z direction were similar, with the values grouped around -6°, -16°, and 29° for subjects one two and three respectively. The Y component was constant for all conditions for all subjects, though actual values differed slightly between subject two at approximately 63° and subjects one and three at approximately 80°.

Elbow: The center of oscillations for the X component of the elbow were relatively constant with respect to task frequency and dynamic level for all subjects. Actual values were in the area of -96° for subjects one and three, and -111° for subject two. Subjects one and three showed similar results for the Z component, with actual values of approximately -132° and -142°. Subject two however showed a slight decrease (indicating medial rotation) of about 30° with increasing dynamic level.

Wrist: The center of oscillation for the X component of the wrist showed a slight increase (from approximately -22° to -10° for subject one, -11° to 7° for subjects two and three) for all subjects as dynamic level increased, indicating extension of the wrist from a flexed position toward a position with the hand in line with the wrist (and moving further from its fixed final position value). All subjects showed a slight decrease in the Y component with increasing dynamic level, indicating adduction of the wrist. Subject one and two had similar values between 13° and -3°, subject three had values between -5° and -15°.

Fulcrum: The center of oscillation for the X component of the fulcrum wrist showed an increase with increasing dynamic level for all subjects. Actual values were also similar for all subjects, and ranged from 15° at a low dynamic level to 53° at a high one. The center of oscillation of the Y component showed a slight decrease with increase in playing dynamic level. Values for subject three (-11° to -21°) were slightly different than values for subjects one and two (-23° to -38°).

The values for center of oscillation of each joint reflect the posture assumed by the subject for the single stroke drumming task, in which the subject is sitting upright, upper-arm hanging naturally beside body with a slight medial rotation, elbow flexed approximately 90° so the forearm is horizontal, and a supinated forearm so the palm is facing downward. The increase in the wrist X component with increasing dynamic level is expected, because as the dynamic level (and consequently ampli-

tude of oscillation) increases, the center of oscillation must move away from the fixed final (impact) position.

Frequency of oscillation

Shoulder: With few exceptions where no oscillation occurred (at low dynamic level and high frequency), the oscillating frequency of the X and Z components matched the playing speed for all subjects. The Y component oscillated only at low a low task frequency, following the playing speed.
Elbow: The elbow X and Z components did not oscillate consistently at low dynamic levels and higher tasks speeds. The oscillating frequency otherwise followed the task frequency.
Wrist: With one exception when no oscillation occurred, the oscillating frequency of both the X and Y components of the wrist joint followed the playing frequency for all subjects and conditions.
Fulcrum: Subject one showed no oscillations at a low dynamic level for either the X or Y components. At a low task frequency, the oscillating frequency of the Y component was twice the task frequency. Under all other conditions, the oscillating frequency of both components followed the task frequency for all subjects.

As expected in this simple drumming task, whenever oscillations occur, they generally do so at the task frequency.

Amplitude of oscillation

Shoulder: The amplitude of oscillation of both the X and Z components showed an increase with increasing dynamic level, and no visible trend with respect to task frequency. Amplitudes ranged from 0° to 6°. The Y component also followed this trend, but to a smaller degree. Amplitudes ranged from 0° to approximately 1.5°.
Elbow: The Oscillating amplitudes of both the X and Z components show a slight increase with dynamic level, and no visible trend with respect to task speed. As with the shoulder, the oscillations were small, in the range of 0° to 3° for subjects one and three, and 0° to 7° for subject 2.
Wrist: The amplitude of oscillation of both the X and Y components showed a general increase with increasing dynamic level for all subjects. Values ranged from 1° to 23° for The X component, and 0° to 10° for the Y component. Amplitude values were similar for all subjects, except for subject three which had larger values for the X component, and slightly smaller values for the Y component.
Fulcrum: The amplitude of oscillation of both the X and Y components showed a general increase with increasing dynamic level. X component values for subjects two and three were similar, and in the range of 0° to 14°. Subject one values were slightly smaller at 0° to 8°. Y component values for subjects one and two were similar, and in the range of 0° to 10°. Subject three values were slightly smaller at 0° to 4°.

It is apparent then, that the X component of the wrist segment provides the majority of the motion for this stroke. Overall, the oscillations in the shoulder and elbow were small, indicating that perhaps

the motion is was not voluntary, and was caused by the inertia of the more active components such as the wrist. The shoulder Y component seldom oscillated, and only did so at low amplitudes indicating this component in not involved (involuntarily or otherwise) in the single stoke.

Relative phase

Shoulder: Phase angle results for the shoulder joint did not yield clear results. Though quite similar, X and Z component phase angles for subjects two and three show no trend or preferred values. Despite this however, the phase difference between the two components was always close to 180°. Subject one in contrast, shows preferred values at higher dynamic levels. With the exception of high task speed at a low dynamic level, the X and Z components are clustered about 321° and 184°. Phase angles for the Y component showed a preferred value of 357° for subject one only.

Elbow: Phase angle results for the elbow joint also yielded inconclusive results. Again, though quite similar, X component phase angles for all three subjects only showed a preferred value of approximately 169° at low playing frequency. No trends or preferred values were otherwise apparent for either component.

Wrist: Phase angles for the both the X and Y components of the wrist showed strong preferred values for all subjects of approximately 328° and 149°respectively. Again the phase difference between theses two components is nearly 180°. furthermore, The relative phase values of the wrist and shoulder (for subject one) were similar.

Fulcrum: Phase angles for the the X and Y components of the fulcrum also showed preferred values across all subjects for moderate and high dynamic levels. X component values were clustered at approximately 275°, and Y component values were clustered at approximately 115°. The phase difference between the two components is 160°, again close to 180°.

For conditions under which preferred values were apparent then, the following relationships can be made: Relative phase of the shoulder X, wrist X, and elbow X components were similar, in were all in-phase with each other. Relative phase of shoulder Z and wrist Y components were similar, and in-phase with each other. The above two groups were anti-phase with each other. Finally, fulcrum joint X and Y components were in anti-phase with each other, and lagged the above two groups by approximately 45°.

5.4 Conclusions

Definite trends were observable in the single strike drumming task. Segment oscillation frequencies followed playing frequencies in nearly all cases. Oscillations amplitudes in the wrist and fulcrum increased with increasing dynamic level, and were accompanied with corresponding center of oscillation changes. The wrist component consistently showed the largest values, indicating the wrist produced most of the motion in this stroke. Center of oscillation values showed preferred values for

the shoulder and elbow segments. Phase values showed strong preferred values in many cases, but also showed seemingly random variability for others.

Chapter 6

Rhythmic motion though Matsuoka neural oscillators

6.1 Introduction

This chapter will introduce the motion primitive used in this work: the Matsuoka neural oscillator. The oscillator will first be described in detail, including past analysis of its properties, and past uses as a motion primitive. Finally, the configuration used in this work will be described including implementation details.

6.2 The Matsuoka neural oscillator

This oscillator was first proposed by Kiyotoshi Matsuoka in [51][52], and describes a model of spinal cord neurons that produce rhythmic motion. It has been used frequently in for producing rhythmic motions in a wide range of applications, including sawing, juggling tennis ball with racket, drumming The Matsuoka oscillator is governed by the following equations.

$$\tau_1 \dot{x}_1 = -x_1 - \beta v_1 - \gamma [x_2]^+ - \sum_{j=1}^{j=n} h_j [g_j]^+ + c \tag{6.1}$$

$$\tau_2 \dot{v}_1 = -v_1 + [x_1]^+ \tag{6.2}$$

$$\tau_1 \dot{x}_2 = -x_2 - \beta v_2 - \gamma [x_1]^+ - \sum_{j=1}^{j=n} h_j [g_j]^- + c \tag{6.3}$$

$$\tau_2 \dot{v}_2 = -v_2 + [x_2]^+ \tag{6.4}$$

$$y_i = [x_i]^+ \tag{6.5}$$

$$y_{out} = y_1 - y_2 \tag{6.6}$$

$$[x]^+ = max(x, 0) \tag{6.7}$$

$$[x]^- = -min(x, 0) = max(-x, 0) \tag{6.8}$$

A schematic of the oscillator is shown in Fig. 6.1. g_i is an input to oscillator, scaled by the constant gain h_i. g_i can be a proprioceptive input, an input from other neurons, or some other periodic

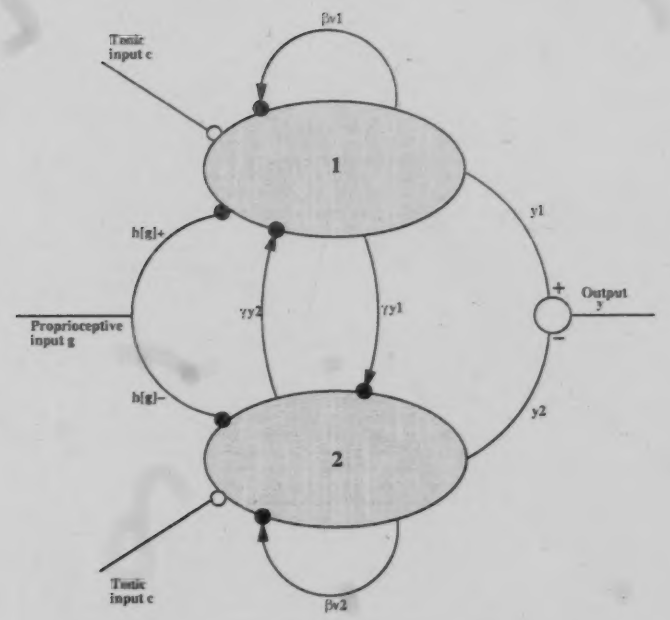


Figure 6.1: Schematic of Matsuoka mutually inhibiting neurons. solid circles represent inhibition.

forcing signal. Through the $[]^+$ and $[]^-$ operators, the input g_i inhibits one neuron while it activates the other. c is a tonic input, and determines the magnitude of oscillation. Furthermore, if c is zero, no oscillations occur. x_i are the firing rates of each neuron. v_i are internal states of the neurons representing self-inhibition. y_i are the outputs of each neuron, and are taken as the positive component of x_i using the $[]^+$ operator. β and γ modulate self and mutual inhibition of the neurons, and are generally constants. Finally, y_{out} is the output of the oscillator, and is the the signal that is used to produce motion. τ_1 and τ_2 are oscillator time constants, and determine the frequency and shape of the oscillation. According to Williamson [87], for stable oscillation τ_1/τ_2 should be in ratio 0.1-0.5. The $[]^+$ and $[]^-$ terms in equations 6.1 and 6.3 are non-linear, so the neural oscillator as a whole is a non-linear system. Figure 6.2 shows a block diagram of oscillator as a dynamic system, composed using the Matlab Simulink toolbox.

6.2.1 Oscillator behavior and properties

The oscillator has several parameters, and its behavior varies with them, as well as with the inputs g_i and c. Following is a summary of the work done to date in describing the behavior of these non-linear oscillators.

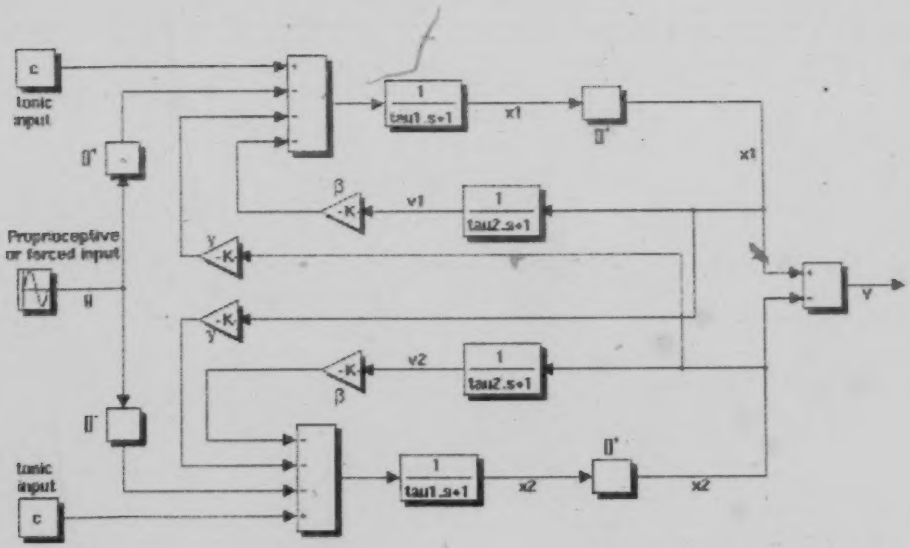


Figure 6.2: Matsuoka neural oscillator as dynamic system.

Free oscillations

Under certain ranges of parameters, the oscillator will vibrate with no input g_i at its natural frequency $w_{n_{osc}}$ and amplitude A_{osc}, determined by τ_1 and c respectively [87][8]. Furthermore, β and γ will affect the shape of the oscillations. Figure 6.3 shows the oscillator output with no input signal g_i, and typical parameters.

Entrainment and forced oscillations

If an oscillating input g_i is applied, and its frequency of oscillation is sufficiently close to $w_{n_{osc}}$, and its amplitude sufficiently large, the oscillator can lock onto the frequency of the input signal. The oscillator will vibrate at the same frequency as the input, but with some phase shift and gain. The oscillator can entrain to an input over a large range with out need for parameter adjustment [89], but the phase and gain will vary as the input frequency varies. Figure 6.4 shows the system at various degrees of entrainment to an input frequency different from $w_{n_{osc}}$. At low input amplitudes, the oscillator is not entrained, and oscillates at its natural frequency. As the input amplitude increases, the oscillator follows the input for a few cycles, then misses a cycle. When the input is sufficiently large, the oscillator locks onto the input signal frequency.

Describing oscillator properties using describing function analysis

Frequency domain methods cannot be applied directly to nonlinear systems, because frequency response functions cannot be defined for these systems [65][8]. Linear analysis techniques therefore

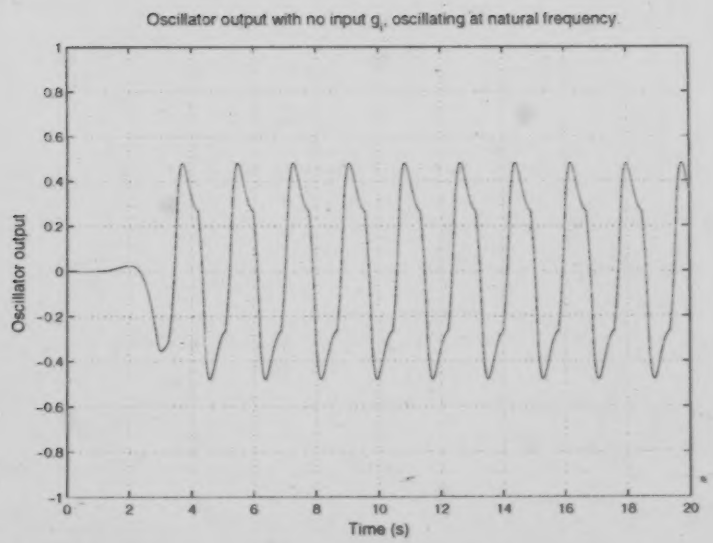


Figure 6.3: Oscillator output with no input, oscillating at its natural frequency and amplitude. Oscillator parameters are: $c = 1$, $\tau_1 = 0.2$, $\tau_2/\tau_1 = 2$, $\beta = 2.5$, and $\gamma = 2.5$, for which the natural frequency and amplitude are $w_{n_{osc}} = 3.53 rad/s$ and 0.48.

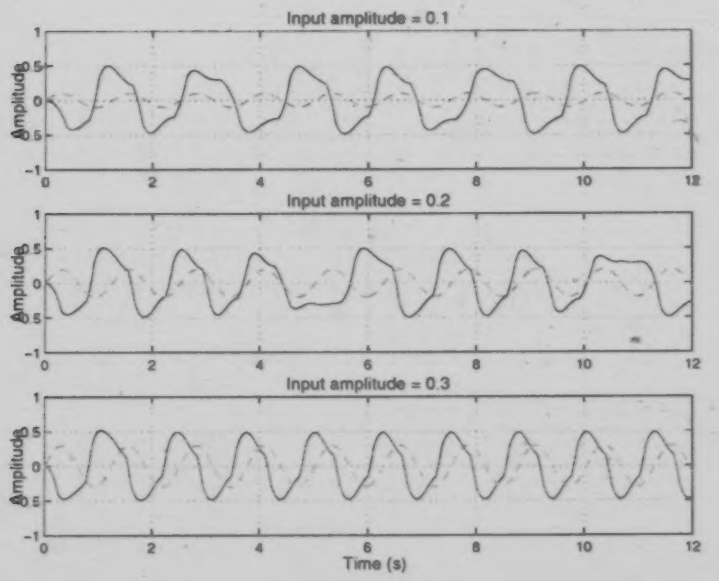


Figure 6.4: Oscillator at various stages of entrainment to an input. Oscillator parameters are: $c = 1$, $\tau_1 = 0.2$, $\tau_1/\tau_2 = 0.5$, $\beta = 2.5$, $\gamma = 2.5$. The corresponding oscillator natural frequency is $w_{n_{osc}} = 3.53 rad/s$, and the input g_i is 5 rad/s.

can not be used for oscillator analysis.

In [86], Williamson proposes using describing function analysis to estimate the oscillator behavior. Arsenio has also used this approach to describe the oscillator behavior [8][6][7]. For describing function analysis to be valid, the non-linear system must must meet 2 conditions: first, the input to the non-linearity must be sinusoidal. Second, the linear components of the overall system following non-linearity must be sufficiently low-pass to attenuate all harmonics to such values as to be negligible [65][8]. If these conditions are not met, describing function analysis will result in large estimate errors [65]. Describing function analysis approximates the steady state output signal of the entire system (linear and non-linear components) as a single frequency Fourier series. The output is approximated as a sinusoid of the same frequency, but with a different amplitude and gain, and the non-linearity can thus be replaced with a complex gain.

Williamson takes a simulation approach to describing function analysis. He evaluates the behavior of the oscillator (with fixed parameters) output (frequency, amplitude, and phase) over a range of input frequencies and amplitudes using a Fourier transform, and neglects points where the oscillator does not entrain the input [88]. The results can be presented in a linear system format such as bode plot, displaying the gain and phase relationships with respect to the input frequency. Figures 6.5 and 6.6 are such plots. Figure 6.5 shows the output amplitude with respect to input frequency, plotted for several values of input amplitude. Figure 6.6 shows the gain and phase of the neuron plotted against input frequency for several values of input amplitude. Plotting this relationship for different values of τ_1 and input frequency range produces plots of the same shape, but shifted along the frequency axis [87]. It is therefor more convenient and useful to plot this relationship with the frequency axis normalized as $w/w_{n_{osc}}$. One conclusion Williamson draws from this analysis, is that since the output amplitude does not vary much with the input amplitude, the sensitivity of the oscillator to the input gain h_i is low.

In [8][6][7], Arsenio instead uses an analytical approach to describing function analysis. According to Arsenio, when the neural oscillator system is entrained to the input g_i, it satisfies the conditions for describing function analysis to be valid. Arsenio derives the describing functions to estimate the oscillator behavior, and develops algebraic equations to determine the oscillator behavior from its parameters and the input function.

From [8], the single frequency Fourier series for an input of amplitude A and bias B to the neuron nonlinearity are

$$N_a(A,B) = \frac{1}{2} + \frac{1}{\pi}\left[arcsin\left(\frac{B}{A}\right) + \frac{B}{A}\sqrt{1-\left(\frac{B}{A}\right)^2}\right] \tag{6.9}$$

$$N_b(A,B) = \frac{1}{2} + \frac{1}{\pi}\frac{A}{B}\left[\frac{B}{A}arcsin\left(\frac{B}{A}\right) + \sqrt{1-\left(\frac{B}{A}\right)^2}\right] \tag{6.10}$$

For natural oscillations, the frequency of oscillation is proportional to τ_1, and output amplitude is

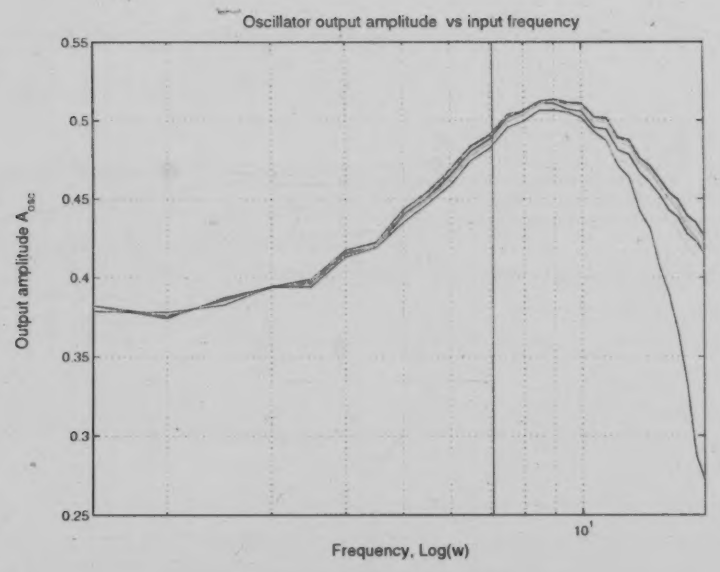


Figure 6.5: Oscillator output amplitude plotted as a function of input frequency for several values of input amplitude. Oscillator parameters are: $c = 1$, $\tau_1 = 0.1$, $\tau_1/\tau_2 = 0.5$, $\beta = 2.5$, and $\gamma = 2.5$. Input amplitudes range from 0.5 to 1.75. The vertical line indicated the natural frequency of the oscillator, $w_{n_{osc}}$. Input amplitude (and therefore h_i) has little effect on the neuron output amplitude.

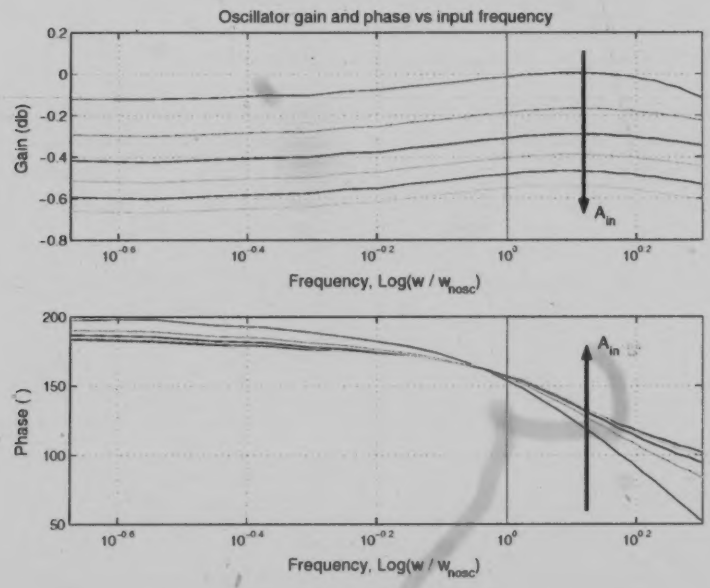


Figure 6.6: Oscillator gain and phase as a function of input frequency. Oscillator parameters are: $c = 1$, $\tau_1/\tau_2 = 0.5$, $\beta = 2.5$, and $\gamma = 2.5$. The results are presented with the frequency axis normalized with respect to $w_{n_{osc}}$. Changing the value of τ_1 produces a plot of the same shape, but shifted on the frequency axis. It is therefore more useful to present them in this manner.

proportional to c as shown below,

$$\omega_{n_{osc}} = \frac{1}{\sqrt{\tau_1 \tau_2}} = \frac{1}{\sqrt{k}\tau_1} \tag{6.11}$$

$$Ay_{out} = 2N_a A \tag{6.12}$$

where, for the condition of natural oscillation

$$N_a = \frac{1+k}{\gamma k}$$

and

$$A = \frac{2c}{\beta + \gamma}.$$

For a forced response, given a sinusoidal input g_i of amplitude D, the oscillator is entrained, and the frequency of oscillation is the same as forced input. The amplitude and bias of the input x_i to the non-linearity are

$$N_a^2 a_l + N_a b_l + c_l - \frac{D^2}{4A^2} = 0 \tag{6.13}$$

and

$$B = \frac{c - \frac{D}{\pi}}{1 + (\beta + \gamma)N_b}, \tag{6.14}$$

where

$$a_l = \left(-\gamma + \frac{\beta}{\tau_2^2 w^2 + 1}\right)^2 + \frac{\beta^2 \tau_2^2 w^2}{(\tau_2^2 w^2 + 1)^2}$$
$$b_l = -2\gamma + 2\beta \frac{1 - \tau_2 w}{\tau_2^2 w^2 + 1}$$
$$c_l = \tau_1^2 w^2 + 1$$

The oscillator output amplitude Ay_{out} is again given by equation 6.12. However, to obtain N_a and A for a forced response, it is necessary to solve for A, B, N_a, and N_b using equations 6.9,6.10,6.13, and 6.14. The phase is given by

$$\phi = \pi = atan2(imag(2N_a L(jw)), real(2N_a L(jw)))$$

where

$$L(jw) = \frac{1}{2} \frac{1}{\tau_1 wj + 1 + (\frac{\beta}{\tau_2 wj + 1} - \gamma)N_a}.$$

A more detailed analysis of the oscillator stability and entrainment properties, and the bounds of describing function analysis are presented in [51][52][87][89] [8][6][7].

6.2.2 Neural oscillators as motion primitives

Neural oscillators are used as motion primitives in a number of ways, but generally always drive a robot in joint space. That is, the output of the oscillator is used to drive the joint of a robot. In

general, the differences in usage are in the source or type of the input g_i. Three methods, or input types most commonly used in humanoid robotics, and are outlined below. Regardless of the method used, the oscillator design problem is to determine the output amplitude or gain and phase given a set of oscillator parameters and an input function; or alternatively, to select the oscillator parameters (and possibly input signal) to obtain a desired gain and phase. Following are some method in which oscillators are used as motion primitives to create robot motion.

Natural Oscillation

The simplest method employs no input signal g_i, but takes advantage of the oscillator's property of oscillating at its natural frequency. τ_1 and c are used to specify the frequency and amplitude of oscillation respectively. This approach is used by Schaal et al. in [78] However, under these conditions, the output of the oscillator is not smooth (for example, Fig. 6.3), and not suitable to produce smooth rhythmic trajectories. Schaal et al. solve this problem by taking the output of the oscillator y_{out} as a joint velocity, and integrating the signal to create a smoother joint position trajectory. A high pass filter is also required with this method to remove the bias introduced during the integration.

Periodic forcing function

Another method uses a periodic (though not necessarily sinusoidal) forcing function, such as a processed audio signal or a pure sine wave as the input g_i, and adjust the oscillator parameters to ensure entrainment takes place. In [89], the oscillator is synchronized to an auditory signal. The input signal g_i is sound captured by microphone, of a periodic auditory event. In this case, the event was a drum being struck. Though periodic, the input signal is not at all sinusoidal (large spikes are generated at impact). The oscillator can still entrain this signal, though over a much smaller range of frequencies. This requires that τ_1 be tuned so $w_{n_{osc}}$ is close to the desired or expected frequency of oscillation. This method also produces an oscillator output which is not smooth. Another disadvantage of this method is that the equivalent gain and phase of the oscillator are not known for this type of input, though the output amplitude is close to that of the free vibration condition, Ay_{out}.

In [79], Schaal et al. also the sound signal from a drumming task as the input g_i. To obtain a predictable gain and phase, Schaal et al. adjust τ_1 so the natural frequency of oscillation $w_{n_{osc}}$ is the same as that of the input g_i. In this manner, the gain and phase of the oscillator are always constant, and tuning becomes simple. However, under these conditions, the output of the oscillator also is not smooth, and integration and filtering are necessary. This method also requires that the frequency or period of the input signal is known.

Connecting to dynamic systems

Yet another method of using neural oscillators as motion primitives involves tightly coupling the oscillator to another dynamic system by using proprioceptive feedback; that is, the output of the oscillator is the desired state (position or velocity) input to the dynamic system, and the actual state of the system (with some feedback gain) is the input g_i to the oscillator. This method tunes into the resonant frequency of the system and exploit its natural dynamics [89][86][87] For the oscillator to drive the dynamic system in resonance, $w_{n_{sys}} \geq w_{n_{osc}}$ [8][89]. The final frequency of oscillation depends on the interaction between the oscillator and the dynamic system. This method requires τ_1 be tuned so the final frequency of oscillation is in a range that the oscillator can entrain. This is not difficult since the oscillator can entrain such a near sinusoidal input over a large range of frequencies without change in oscillator parameters. The disadvantage of this method is that since the final frequency of oscillation, and therefore the input frequency to the oscillator, is not straight forward to determine, the gain and phase of the oscillator are difficult obtain beforehand. This method has been used to drive the arms of a humanoid robot to produce several types of behavior. Though interesting, this method can not drive a system at a desired frequency, but only drive the dynamic system in in resonance.

6.2.3 Networks of oscillators for multi-joint control

In any practical application using neural oscillators as motion primitives to create motion, motion of more than one joint is usually involved, and a number of oscillators are required (typically one per joint). The problem then arises of how to organize the oscillator(s) to achieve the desired motion. Specifically, the challenge is in controlling the phase relationships of the involved oscillators. Two common methods of connecting networks of oscillators are described below. A fundamental difference between these two methods is that one emphasizes a hierarchy, while the other does not.

Mutually connected networks

In this configuration, the output of each neuron is connected to other neurons in an inhibiting manner. For example, given two neural oscillators O_1 and O_2, each made up of two neurons ($_a$ and $_b$), the input to oscillator one becomes

$$InputO_{1,a} = \sum_i h_i[g_i]^+ + h_c\alpha[x_{2a}]^+ + h_c(1-\alpha)[x_{2b}]^+$$

$$InputO_{1,b} = \sum_i h_i[g_i]^- + h_c\alpha[x_{2b}]^+ + h_c(1-\alpha)[x_{2a}]^+$$

and the original oscillator equation 6.1 and 6.3 become

$$\tau_1 \dot{x}_1 = -x_1 - \beta v_1 - \gamma[x_2]^+ - \sum_{j=1}^{j=n} h_j[g_j]^+ - h_c\alpha[x_{2a}]^+ - h_c(1-\alpha)[x_{2b}]^+ + c$$

$$\tau_1 \dot{x}_2 = -x_2 - \beta v_2 - \gamma[x_1]^+ - \sum_{j=1}^{j=n} h_j[g_j]^- - h_c\alpha[x_{2b}]^+ - h_c(1-\alpha)[x_{2a}]^+ + c$$

Here, α modulates the relative phase between the two oscillators, where $\alpha_i \in [0, 1]$. $\alpha = 0$ and $\alpha = 1$ produce phases of 0 and π respectively between the two oscillators. Intermediate values of α produce intermediate phase values which are not straight forward or intuitive, and which are sensitive to parameter changes and initial conditions. Consequently phase differences other than $\pm\pi$ are difficult to achieve, and this results in a systems that is difficult to tune [89].

Master-slave networks

In this configuration, shown in Fig. 6.7, one oscillator acts as a master oscillator, and its output is connected in inhibition to n number of slave oscillators. The connections are unilateral, so the slave oscillators have no effect on the master oscillator. Again, given master and slave oscillators, O_m and O_s respectively, each made up of two neurons (a and b), oscillator one acts as the master, the inputs to the slave oscillator become

$$InputO_{s,a} = -A_i\left(\alpha_n\left[x_{ma}\right]^+ + (1-\alpha_n)\left[x_{mb}\right]^+\right) \tag{6.15}$$

$$InputO_{s,b} = -A_i\left((1-\alpha_n)\left[x_{ma}\right]^+ + \alpha_n\left[x_{mb}\right]^+\right) \tag{6.16}$$

where A is a coupling weight, and again the value of α modulates the phase. The same input can be used for n number of slave oscillators. This approach was suggested by Schaal et al. in [78], though no mention was made on the difficulty in achieving intermediate phase values. In fact, as with the previous method, with this method it is also very difficult to reliably obtain intermediate values of phase. Furthermore, the range of phases from $0°$ to $-\pi°$ is not continuous. Because the connection weights are unreliable in producing phase values, Williamson instead suggest replacing the slave oscillators with fixed gains [89]. This limits the possible phase values to $0°$ and $\pm\pi°$ (using negative gains). Williamson employs this method to make a humanoid robot saw a piece of wood.

6.3 Methods

For this work, the oscillators were used in a configuration that allowed them to respond in similar manners to each of the three input types described above. No input, a forcing function, or a proprioceptive feedback input could be used. For the case of no input, τ_1 was adjusted so $w_{n_{osc}}$ was equal to the desired frequency of oscillation. When connecting to a dynamic system and using a

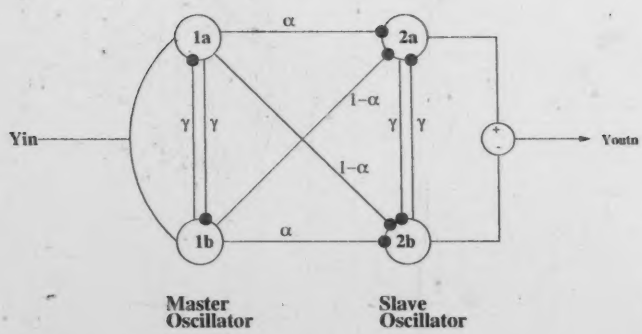


Figure 6.7: Master-slave oscillator configuration. Neurons $1_{a,b}$ and $2_{a,b}$ represent the master and slave oscillators respectively. The connections are unilateral, so the slave oscillator has no influence on the master.

proprioceptive feedback, the end-effector trajectory, or, as Williamson suggest in [89], a sum of normalized joint trajectories can be used as the input g_i. When using forced or proprioceptive inputs, τ_1 must be adjusted such that $w_{n_{osc}}$ is close enough to the final oscillating frequency to cause entrainment. $w_{n_{osc}}$ can be controlled by adjusting τ_1 using equation 6.11. As mentioned previously, when $w_{n_{osc}}$ is close or equal to the oscillating frequency, the oscillator output is not smooth (see Fig. 6.3). To ensure a smooth position-trajectory for robot control, the oscillator output was treated as a joint velocity, and integrated, and a low pass filter was used to eliminate the bias introduced by the integration. Figure 6.8 shows the output of such the used oscillator with proprioceptive feedback.

For multi-joint control, the master-slave network suggested in [78] was used, with a slight modification, to allow for an even and continuous range of phases, and thus create the necessary phase relationships between oscillators. In addition to two inputs in equations 6.15 and 6.16 used to control the phase relationship of the master and slave oscillators, the output of master oscillator was also used as the input g_i of the slave oscillator. Although this configuration causes a great deal of internal tension in the oscillator, it succeeds in creating a continuous range of phases between the 2 oscillators in the range of $[0°, -\pi°]$. This internal tension induced in the slave oscillator can be seen in Fig. 6.9: the oscillator begins at some initial phase relative to the master oscillator, but is then forced into a different phase by the combination of inputs. Figure 6.10 shows the slave oscillator's outputs at different phase values, corresponding to α values of 0, 0.3, and 1. A disadvantage of this method in obtaining phase relationships between oscillators is that the coupling weights α affect the output amplitude of the oscillator, and thus the oscillators amplitude (and gain) is now dependent on two variables: c and α. Since the above method can only create relative phases of $[0°, -\pi°]$, another mechanism was needed to to obtain a full $[0°, -2\pi°]$ phase range. This was accomplished by introducing one more parameter, k, which simply multiplies the slave oscillator output by 1 to obtain a phase range of $[0°, -\pi°]$, or -1 to obtain a phase range of $[-\pi°, -2\pi°]$.

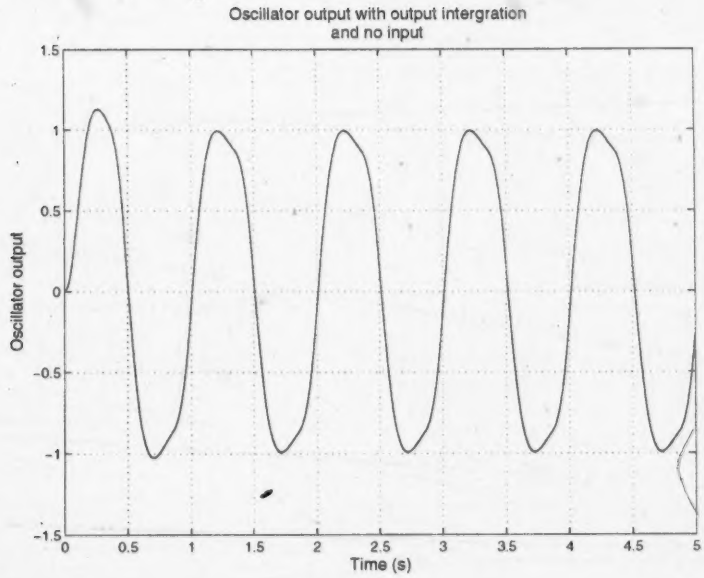


Figure 6.8: Master-slave oscillator configuration. Oscillator parameters are $\tau_1 = 0.1125$, $\tau_2 = 2\tau_1$, $c = 2.8$, $\beta = 2.5$, and $\gamma = 2.5$.

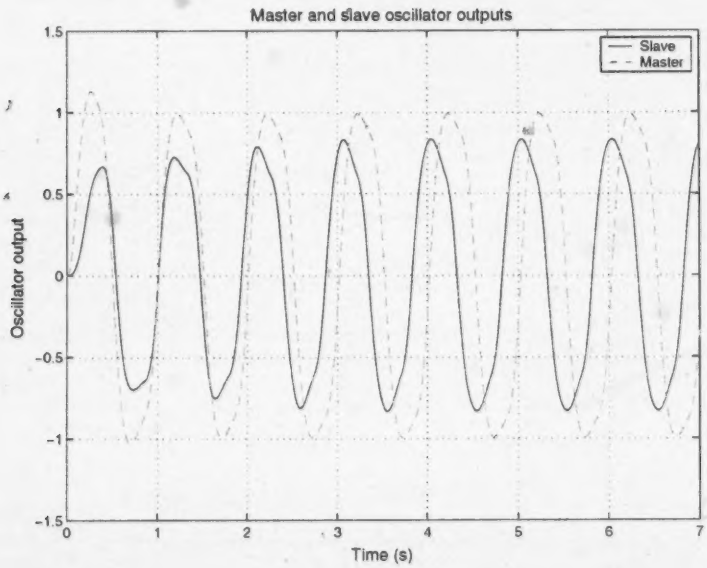


Figure 6.9: Trace showing the internal tension in slave oscillator. In the first two cycles of oscillation, the slave oscillator tends toward an in-phase relationship with the master oscillator. As the oscillations progress, the slave is forced to a relative phase of roughly $-\pi/2$. Oscillator parameters are $\tau_1 = 0.1125$, $\tau_2 = 2\tau_1$, $c = 2.8$, $\beta = 2.5$, $\gamma = 2.5$, and $\alpha = 0.34$.

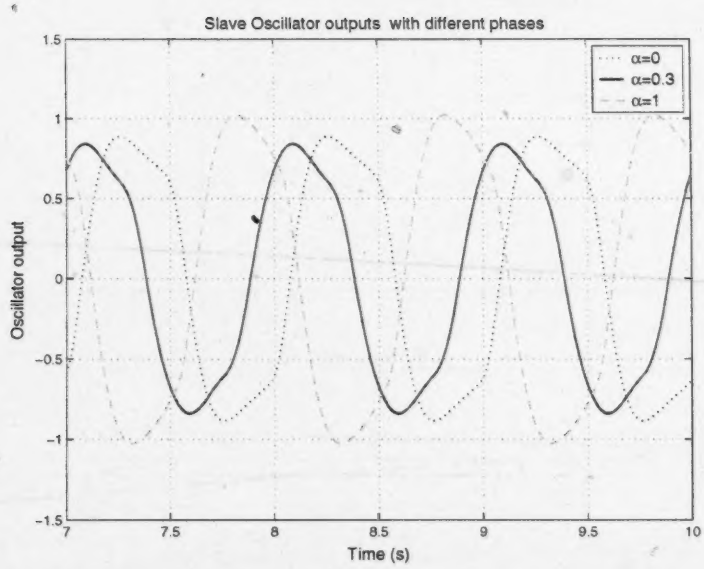


Figure 6.10: Master-slave oscillator outputs at various values of relative phase. Values of α are 0, 0.3, and 1, and correspond to phase values of approximately 0, $-\pi/2$, and $-\pi$ respectively. Oscillator parameters are $\tau_1 = 0.1125$, $\tau_2 = 2\tau_1$, $c = 2.8$, $\beta = 2.5$, and $\gamma = 2.5$.

Figures 6.11, 6.12, and 6.13 show the individual master and slave oscillator systems, and the final master-slave configuration respectively. In total then, there are four parameters with which to tune the oscillator to obtain a desired output. τ_1 controls the frequency of oscillation, c controls the amplitude of oscillation, α controls the phase of oscillation and also affects the output amplitude, and k controls the range of the phase shift values controlled by α.

Implementation

All calculations and computations were performed using Matlab. Neuron models and calculations were done using the Simulink toolbox, using a fixed step size of 0.0125 s for all simulations (which corresponds to an 80 Hz sampling rate). Also, for subsequent simulations, no-input was used for g_i.

Conclusions

Though modulation of the Matsuoka oscillator behavior is possible, it is not straight forward because of the number and interaction of the parameters. Specifically, two parameters are needed to modulate phase, and one parameter, α, modulates both phase and amplitude. Though functional, this oscillator is not ideal as a motion primitive.

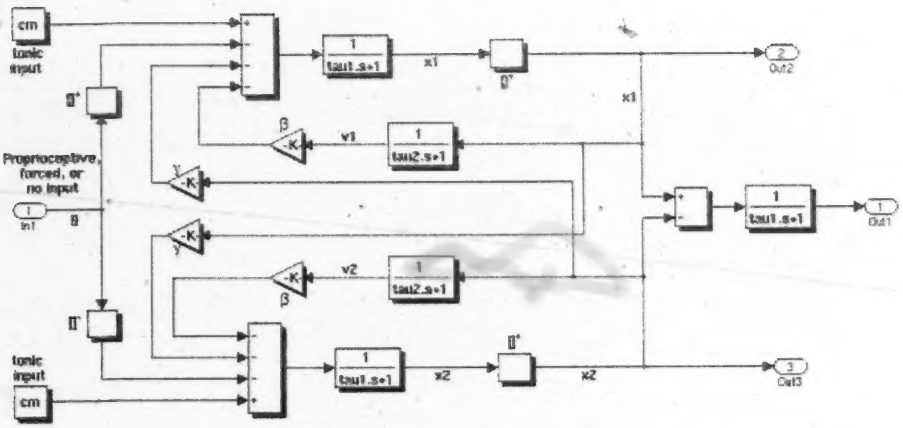


Figure 6.11: Master oscillator Simulink model.

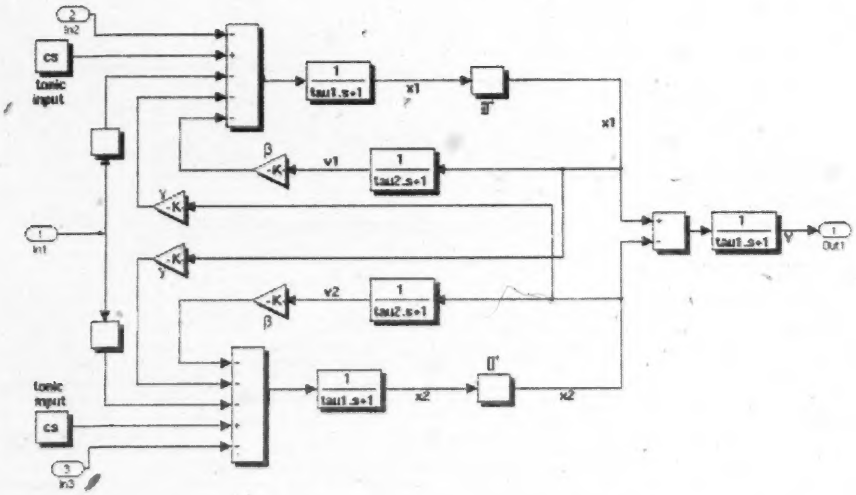


Figure 6.12: Slave oscillator Simulink model.

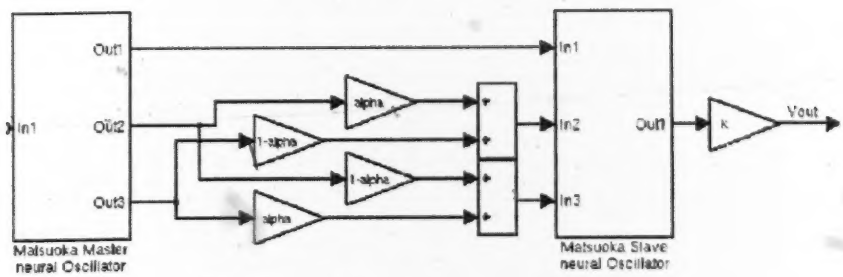


Figure 6.13: Master-slave configuration Simulink model.

Chapter 7

Imitation learning

7.1 Introduction

In this chapter, the task of learning a particular behavior is addressed. First, the control policy used in this work is discussed. Next, an overview of the learning mode and variables is discussed. Finally, implementation details and results of learning a rhythmic task are presented.

7.2 Methods

7.2.1 Non-linear oscillators as control policies

To encode the learned behavior, the Matsuoka neural oscillator described in chapter 6 was used as the control policy. The master-slave network was used to coordinate the inter-DOF requirements of the movement, requiring a total of 10 oscillators to learn the drumming tasks: nine slave oscillators (one for each DOF), and one master oscillator to control the slaves. The oscillators were configure with no input at g_1.

With this configuration, a total of four oscillator parameters must be adjusted at each slave oscillator to modulate its output to produce an appropriate overall behavior. However, one of these could be determined directly, so only three parameter-values had to be determined though an iterative learning process. τ_1 controlled the speed of oscillation. Since the speed of oscillation was known from the movement analysis (chapter 5), this parameter could be set directly. Furthermore, this parameter was set to be equal for all slave oscillators, as well as for the master oscillator, so all oscillators oscillated at the same frequency. The correct frequency was easily obtained by adjusting τ_1 using equation 6.11. α and k were used to adjust the phase and range of phase of the oscillator output respectively. Since α also affects the oscillator output amplitude (an undesirable side-effect), c had to be adjusted to obtain the correct output amplitude. Learning then consists of determining the correct parameter values, α, k, and c, for each slave oscillator to produce an overall behavior which, as closely as possible, matched the behavior being learned.

Because the output of the oscillator is unit-less, it must be scaled to some meaningful value.

Since the output of the oscillators were intended to drive a robot in joint coordinates, the correct dimension and units were angles in degrees. Consequently, the output of each slave oscillator was multiplied by the amplitude of oscillation calculated in section 5 for the DOF to which the particular slave oscillator corresponded. This, in effect forced the optimization procedure to produce an oscillator output close to unity. Also, since the oscillator output has no bias (that is, it's center of oscillation is about zero), the center of oscillation calculated in section 5 was added to the output of the corresponding oscillator. Figure 7.1 encapsulates the complete control policy structure used for learning of a rhythmic task. The frequency of oscillation of the master oscillator (and thus of all slave oscillators) is set with τ_1. The output amplitude and phase of each slave is set using α, k, and c, and then scaled appropriately to the amplitude A, and bias B for each DOF.

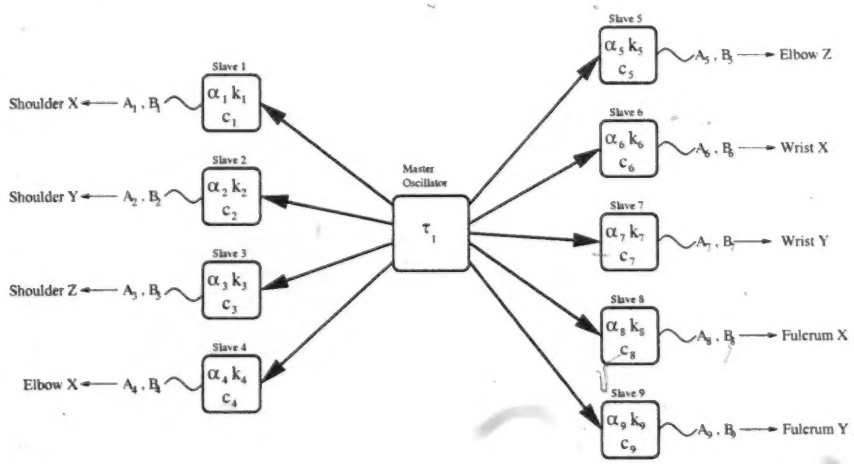


Figure 7.1: Control policy setup for learning of rhythmic task for manipulator with nine DOF's. The master oscillator control the frequency of all slave oscillators. The parameters c, α, and k determine the output amplitude and phase of the oscillator, which is then scaled and adjusted by the calculated amplitude and center of oscillation.

7.2.2 Learning of drumming task

In the type of task being learned in this work, both goal-specific and kinematic components are important. That is, when learning a drumming task, it is important to consider both the actual goals of the task (such as drumming frequency, force, and drumstick tip position and orientation at impact), and the methods used by the teacher to accomplish the task (for example, the actual trajectories of each DOF). To ensure the actual task or goal behavior is learned, an optimization function which contains goal space variables is necessary. Goal space variables would include drumming impact period, force, and stick tip Cartesian position and orientation at impact. To ensure the underlying method used by the teacher is not lost, direct policy learning variables are also necessary in the objective function to preserve the kinematic trajectories of the teacher. These variable can include

the individual joint angles, as well as the drum stick trajectory in Cartesian coordinates. Since it is ultimately more important in learning a task that the goal-space variables are achieved, a weighting scheme can be uses that places more importance on the goal-space error signals than the kinematic ones. With this learning framework, the underlying method used by the teacher is maintained, but adjusted for the students kinematic and dynamic configuration, to ensure the behavior goal is accomplished. Alternatively, instead of attempting to satisfy goal and kinematic variables simultaneously, the kinematic trajectories alone can be learned first (as in direct policy learning), and subsequently used as a starting seed for a goal-space based optimization. In this frame-work, learning would occur on the joint angle trajectories would occur to achieve the desired behavior. One perhaps not obvious advantage of this learning strategy (specifically the drumstick trajectory) is that the small errors in joint angle estimation (discussed in section 4.5) can be corrected for, and thus inverse kinematics re-targeting becomes unnecessary [16].

Ideally, this type of hybrid learning strategy would be pursued to learn the previously described drumming task. However, the time restrictions of this project prohibit dynamic simulation of the drumming task. As a result, a key goal-space variable, the reproduced drumming impact force, is not available as a variable for an optimization function. Consequently, this research will focus on only one component of the hybrid learning strategy: direct-policy learning. Though not such a powerful learning strategy on its own, direct policy learning is an essential part of the hybrid learning strategy, as learning of joint trajectories reduces the solution search space by providing a good seed for the optimization process [76]. Thus, direct-policy learning is till a relevant topic to pursue.

Joint trajectory learning

Joint trajectory learning consisted of determining the correct parameter values, α, k, and c, for each slave oscillator to produce an oscillator output which best matched the joint angle trajectory being learned. The error signal between the recovered $\theta_{recovered}$ (from chapter 4) and the trajectory reproduced by the neural oscillator $\theta_{recreated}$ was simply

$$e_{\alpha,k,c} = \theta_{recovered} - \theta_{recreated}, \tag{7.1}$$

and was a function of the three parameters being determined, α, k, and c. The optimization or criterion function was then

$$argmin_{\alpha,k,c} \sum_{trajectory} (e(\alpha,k,c))^2. \tag{7.2}$$

This optimization procedure was carried out for each DOF of each joint.

7.3 Implementation

All calculations were implemented in Matlab, and the Simulink toolbox was used for neural oscillator computations (as described in chapter 6). Again, the Matlab function lsqnonlin was used for the

optimization task. Because Matlab is an interpreted language, the optimization task ran extremely slow, even on a 1.7 GHz Pentium 4. To slightly relieve this problem, the learning task was performed over a 5 second segment of the drumming task data instead of the entire 30 second capture duration. Even with this restriction, the optimization task took approximately 4.5 minutes to converge for each DOF.

7.4 Results

To analyze the performance of the methods used, data from each of the three subjects performing one drumming trial each was used. The drumming task used for all subjects was the single stroke task at 120 bpm and a moderate dynamic level. Similar to chapter 4, the magnitude of the error between the measured and reproduced (from the learned control policies) stick tip trajectories were used as a measure of the method's overall accuracy in reproducing the learned motion. These results were put into perspective by also comparing them to the stick tip trajectory reproduced from the recovered joint angle trajectories (from chapter 4). Because the motion capture data was acquired at 240 Hz, and the neural oscillator computations were performed at 80 Hz, sub-sampling of the motion capture data by a factor of three was necessary for comparison of the two results. Also for this analysis, for DOF's that showed no oscillation (see section 5 for analysis of motion results), and thus for which no learning took place, the joint angle values were set to the center of oscillation values determined in section 5. Finally, to avoid unnecessary sources of error, the estimated segment lengths were recalculated for this trial, and used in the subsequent analysis. An example of an original (recovered) and reproduced (by the control policy) trajectory, as well as the resulting oscillator parameters for subject one are presented in Fig. 7.2. All original and reconstructed joint angle trajectories for this task, for all subjects are available in appendix B.

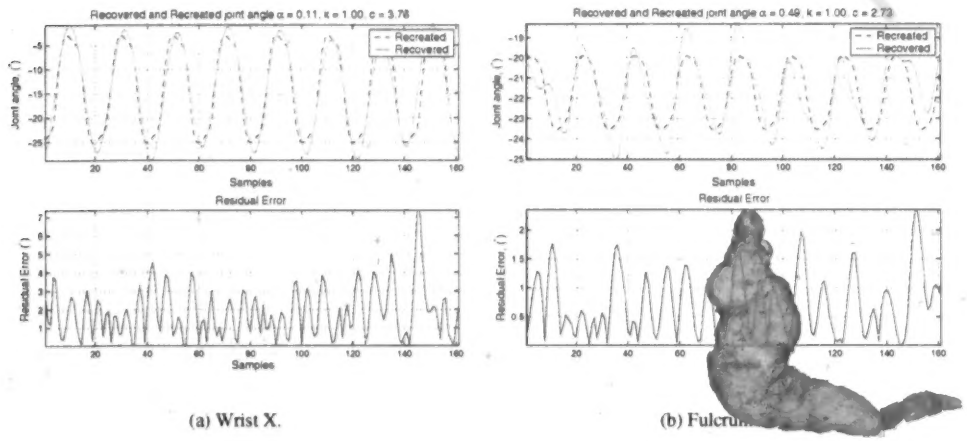
Table 7.1 summarizes the measured task parameters (from chapter 5), and the resulting oscillator parameters from learning of this task [1]. As the task frequency, and consequently oscillating frequency of all components (that showed oscillations) was 4 Hz, the resulting oscillator τ_1 values for all components were equal at 0.028. The resulting oscillator c values reflect the learning tasks' tendency to correct $\alpha's$ influence on the oscillator amplitude, and produce an output of unity. Comparing the measured phase values and the resulting oscillator phase parameters, several relationships can be seen. First, when two components are approximately -180° out of phase (for example the wrist X and Y components for all subjects, and the shoulder X and Z components for subjects one and three), the corresponding α values are similar, but the k values are are negatives. Similarly, for components that have similar phase values (such as the shoulder Z and wrist Y components, and shoulder and wrist X components for subject one, and the elbow X and Z components for subject three) corresponding α values are similar, and k values are equal. Though the oscillator output cap-

[1] The phase values were obtained by subtracting the calculated phase values for the shoulder X component from all other segments and components, and show the relative phases of the components involved in this stroke. These original values are

	X		Y		Z	
	Calculated Frequency, amplitude, phase	Oscillator τ_1, c, α, k	Calculated Frequency, amplitude, phase	Oscillator τ_1, c, α, k	Calculated Frequency, amplitude, phase	Oscillator τ_1, c, α, k
Subject one						
Shoulder	4 Hz, 3°, 0°	0.028, 4.04, 0.17, 1	no oscillation	no oscillation	4 Hz, 3°, -180°	0.028, 3.77, 0.15, -1
Elbow	4 Hz, 1°, -8°	0.028, 2.56, 0.27, 1	N/A	N/A	4 Hz, 1°, -50°	0.028, 2.22, 0.23, 1
Wrist	4 Hz, 8°, -6°	0.028, 3.76, 0.11, 1	4 Hz, 4°, -178°	0.028, 3.76, 0.16, -1	N/A	N/A
Fulcrum	4 Hz, 4°, -56°	0.028, 3.58, 0.38, -1	4 Hz, 2°, -211°	0.028, 2.73, 0.49, 1	N/A	N/A
Subject two						
Shoulder	4 Hz, 2°, 0°	0.028, 3.81, 0.25, 1	no oscillation	no oscillation	4 Hz, 2°, -198°	0.028, 3.5, 0.04, -1
Elbow	N/A	N/A	N/A	N/A	4 Hz, 2°, -284°	0.028, 3.57, 0.42, 1
Wrist	4 Hz, 6°, -74°	0.028, 3.36, 0.45, -1	4 Hz, 5°, -259°	0.028, 3.36, 0.43, 1	N/A	N/A
Fulcrum	4 Hz, 4°, -105°	0.028, 3.66, 0.36, -1	4 Hz, 4°, -272°	0.028, 3.57, 0.39, 1	N/A	N/A
Subject three						
Shoulder	4, 5, 0°	0.028, 3.37, 0.32, 1	N/A	N/A	4, 5, -187°	0.028, 3.40, 0.31, -1
Elbow	4, 1, -26°	0.028, 1.93, 0.25, 1	N/A	N/A	4, 2, -26°	0.028, 2.36, 0.27, 1
Wrist	4, 14, -54°	0.028, 3.16, 0.25, 1	4, 3, -234°	0.028, 3.13, 0.25, -1	N/A	N/A
Fulcrum	4, 5, -88°	0.028, 3.14, 0.04, 1	4, 2, -249°	0.028, 3.42, 0.24, -1	N/A	N/A

Table 7.1: Measured task parameters, and oscillator parameters resulting from learning task. Of interest are the apparent relationship between the measured phase values, and resulting oscillator parameters which control phase, α and k.

tures the essence of the joint motion - its phase, frequency, and amplitude - it does not capture the shape of the oscillations, the slight drift in center of oscillation, or the inter-cycle variations visible in most traces. Thus, in most cases there is considerable error between the original and learned policy. When using direct policy learning as the first step of a hybrid learning strategy, small errors in the control policy are tolerable. However, a major problem, and disadvantage of this particular control policy is the oscillator's inability to alter its topology to more effectively learn the desired behavior.



(a) Wrist X.

(b) Fulcrum

Figure 7.2: Partial time series of recovered and reproduced joint angle trajectories of single stroke drumming task, subject three.

As mentioned previously, the magnitude of the error between the measured and reconstructed (from the learned control policies) stick tip trajectory was used as a measure of the method's overall accuracy in reproducing the learned motion, and was compared to the stick tip trajectory reproduced from the recovered joint angle trajectories. When reconstructing the stick tip trajectory from the learned control policies, oscillations in the torso and collar segments were ignored, and these joint angles were set to the center of oscillation values determined in section 5. The resulting stick tip trajectories and residual errors for a two second section of the single stroke task for subjects one, two, and three can be seen in Figs. 7.3, 7.4, and 7.5 respectively, and are summarized in Table 7.2.

The mean, standard deviation, and maximum error values for both the recovered and reconstructed cases are consistent for all three subjects. All values are considerably larger for the reconstructed case however, about five times larger. The mean reconstruction error across all subjects is 16.4 mm. The mean recovered trajectory error is 3.5 mm In section 4, the human impact coordinate standard deviation was used as an indicator of the typical accuracy achievable by humans in this type of task. For this particular task the mean STD values across all subjects was 5.7 mm. Comparing the reconstructed trajectory error to the results of the subjects' accuracy for this task, we see the mean

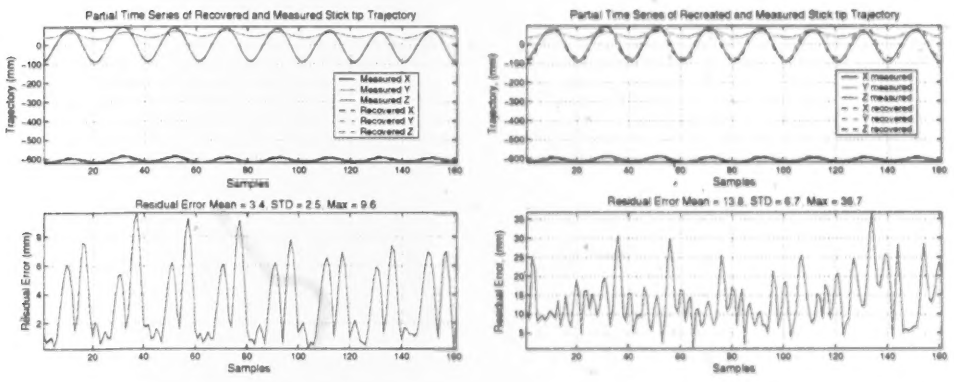
available in Appendix A, Fig. A.4

reconstructed trajectory error is three times that of the variability shown by humans in the same task.

The observed and reconstructed stick tip trajectories were also compared by examining their frequency components (using the methods described in chapter 5). The phase values were calculated relative to the Z component. Figure 7.6 and table 7.3 show the result of the drumstick tip frequency analysis. For all degrees of freedom and all subjects, the observed and recreated trajectories have the same frequency of oscillation. The amplitude values were closely matched, with the maximum error for all cases being 7°. The relative phase values show considerably more variation, with the maximum relative phase error being 25°.

	mm		% kinematic chain	
	Recovered	Reconstructed	Recovered	Reconstructed
Subject one				
Mean	3.4	13.8	0.23	0.93
STD	2.5	6.7	0.17	0.45
Max	9.6	36.7	0.65	2.47
Subject two				
Mean	3.6	19.3	0.28	1.49
STD	2.3	10.7	0.18	0.83
Max	10.4	54.9	0.81	4.25
Subject three				
Mean	3.5	16.1	0.25	1.17
STD	2.2	9.1	0.16	0.66
Max	9.4	45.9	0.68	3.33

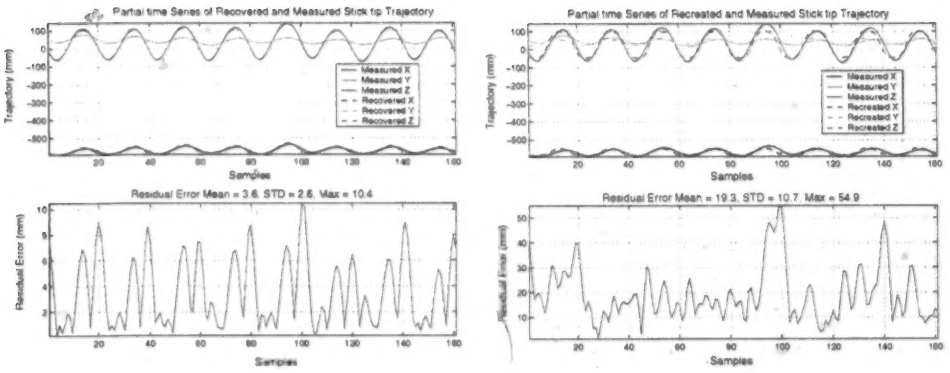
Table 7.2: Resulting segment lengths, and joint center residual errors.



(a) Recovered Stick Tip Trajectory.

(b) Reproduced Stick Tip Trajectory.

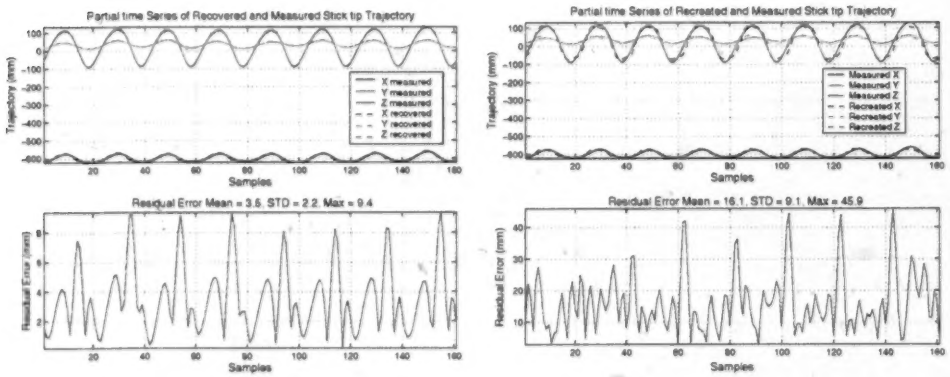
Figure 7.3: Recovered and reproduced stick tip trajectories, and residual error of two second section of single stroke drumming task, subject one.



(a) Recovered Stick Tip Trajectory.

(b) Reproduced Stick Tip Trajectory.

Figure 7.4: Recovered and reproduced stick tip trajectories, and residual error of two second section of single stroke drumming task, subject two.



(a) Recovered Stick Tip Trajectory.

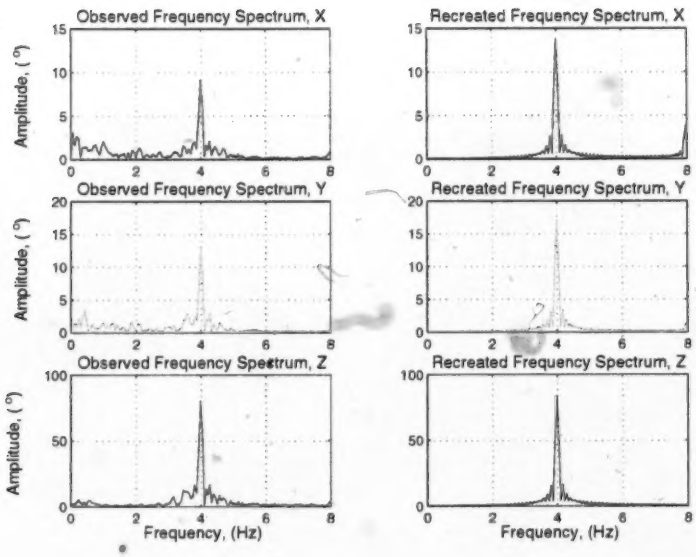
(b) Reproduced Stick Tip Trajectory.

Figure 7.5: Recovered and reproduced stick tip trajectories, and residual error of two second section of single stroke drumming task, subject three.

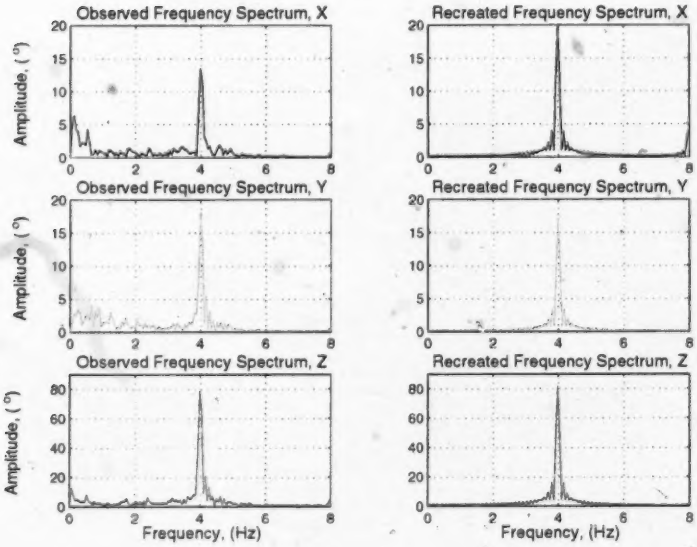
	Subject 1			Subject 2			Subject 3		
DOF	Freq	Amp	Phs	Freq	Amp	Phs	Freq	Amp	Phs
X	4/4†	9/14	-20/-3	4/4	13/20	-6/6	4/4	20/24	-13/-1
Y	4/4	13/17	-19/-1	4/4	18/17	-8/17	4/4	17/21	-7/16
Z	4/4	80/84	0/0	4/4	79/81	0/0	4/4	95/101	0/0

† Observed/Recreated

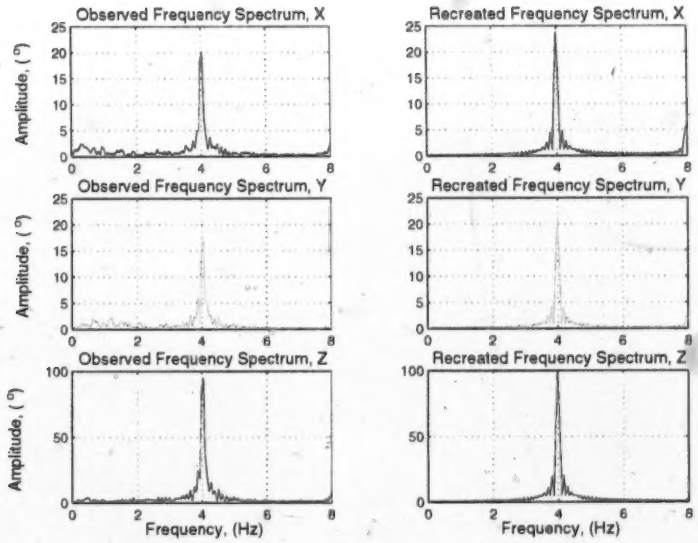
Table 7.3: Frequency, amplitude, and phase of oscillation of observed and recreated drumstick tip trajectory.



(a) Subject 1.



(b) Subject 2.



(c) Subject 3.

Figure 7.6: Frequency analysis results of observed and reconstructed drumstick tip trajectory.

7.4.1 Sources of Error

In addition to the sources of error mentioned in section 4, two additional sources introduced during the learning and reconstruction tasks account for the majority of the observed errors between the reconstructed and original stick tip trajectories. The amount of error introduced by each source can be improved upon.

Choice of control policy

As stated previously, although the oscillator output captures the essence of the joint motion it does not capture its details, such as its shape, and the slight variations in the oscillatory pattern. The particular choice of control policy, in this work Matsuoka neural oscillator, is responsible for the bulk of the introduced error. First, the actual attractor landscape (that is, the shape of the oscillations of the control policy) does not closely match the observed oscillations. This can be seen in Figs. 7.3, 7.4, and 7.5, and in appendix B. The attractor landscape is an intrinsic property of the Matsuoka neural oscillator, and cannot be easily modified.

Drifts in center of oscillation and inter-cycle variations in amplitude could be captured by employing a control policy that included a number of neural oscillators operating at different frequencies and amplitudes, but automatically determining the minimum number of oscillators required to adequately encode the motion is not facilitated by this oscillator. Furthermore, an additional problem is deciding how much detail is adequate, and when increasing the complexity of the control

policy will not yield appreciable improvements in performance.

Neglecting small oscillations

Part of the implementation details of this component of the work included neglecting small (less than 1° amplitude) oscillations in the involved limbs, and all oscillations in the torso and collar segment. The reasons for this were two fold: As the amount of data involved was large, and the simulation environment was inherently slow-running, the more pragmatic reason for this decision was to reduce simulation times. The second, and perhaps more important reason, was that an aim of the work was to encode only active motion, that is, willful motion by the subject that was necessary in producing the tasks in question. It was not desired to encode motion that was simply produced by the inertia of the active limbs. Because humans human have limits in the resolution of active movement, neglecting oscillations less than 1° was a simple way of achieving this goal. The errors introduced by neglecting small oscillations are small compared to those introduced by the control policy.

7.5 Conclusions

As a motion primitive for imitation learning, the Mastuoka oscillator is less than adequate. The reasons for this are two fold: First, as mentioned in chapter 6, modulation of the oscillator behavior is not straight forward because of the number and interaction of the parameters. In addition to this, the oscillator is not capable of altering it topology to closely match the observed behavior. A different choice of motion primitive, one that lacks the above characteristics would thus be advantageous for imitation learning.

Also, direct policy learning alone is adequate only when kinematic trajectories are to be learned, and no other goal-space variables exist. When the task to be learned involves interactions with other objects, or when task dynamics are to be considered, other learning methods should be employed.

Chapter 8

Conclusions and Further Work

8.1 Conclusions

A primary component of imitation learning is the perception and measurement of motion, and all other variables that are important to the overall behavior that is to be learned. This includes raw 3-D coordinates of features of interest (be they on the performer, objects manipulated by the performer, or otherwise), subject postural information when doing model based capture, and other pertinent variables such as force. Passive optical motion capture systems, combined with the methods described to extract kinematic data from the raw motion capture data, are an effective and powerful method of providing this information. Though extremely costly, motion capture systems such as the one used in this work are capable of very accurate 3-D spatial measurements. These systems are are also flexible in the types of subjects they can measure motion of (such as humans, animals, or generally any object to which passive retro-reflective markers can be attached). They also facilitate the integration and synchronization of external data sources often necessary to measure other important goal specific variables. Likewise, the described methods for obtaining kinematic data from raw motion capture data are flexible in that a body model can be created to suit the kinematic structure of any subject. Despite poor joint center estimation methods and results, good overall results were still obtained, as the remainder of the methods used to extract kinematic data from motion capture data were reasonably accurate in reconstructing the measured motion, and were robust to poor subject joint center estimations. Indeed, the systems overall accuracy is largely limited by the complexity of the developed body model. Furthermore, when the kinematic data is used as a seed for imitation learning, the described methods certainly provide sufficient accuracy for a good initialization to the optimization process. The described methods also have the capability to deal with the correspondence problem, as the developed body model can be made to match the body of the student, the model imposed on the body of the teacher, and motion capture marker locations selected to satisfy the body model requirements. Motion capture systems then, combined with the described methods of extracting kinematic data from motion capture data are powerful, flexible, and accurate tools for robot imitation learning.

Throughout the entire imitation learning process, from perception, to reproduction of the learned movement, the major major source of error and bottleneck in accuracy was during the control policy learning. As mentioned previously, the error here was due to the inability of the motion primitive (the Matsuoka neural oscillator) to alter its topology to match that of the observed behavior. Certainly, as a control policy, the oscillator is capable of capturing the major features of the joint angle trajectories, such as their amplitude, frequency, and phase. However, it fails to capture the details of this particular observed behavior, most importantly the shape of the oscillations. This highlights the importance of the choice of motion primitive in a control policy when attempting to learn specific behaviors, and strongly suggests one of the following constraints be met: either the motion primitive must be capable of readily altering its topology to match that of the learned behavior (or facilitate a structure the allows this, such as using multiple motion primitives in a control policy), or that several motion primitives, each with distinct topologies must be available to a control policy to best match the behavior to be learned. Using the Matsuoka oscillator does not satisfy the former constraint, as it can neither alter its topology, nor facilitate a structure that allows this. Furthermore, in the configuration used in this work, it's behavior cannot be modulated in a straight forward manner. Thus, for the particular joint angle trajectories encountered in this work the Matsuoka neural oscillator was a poor choice for control policy motion primitive.

Though not a powerful imitation learning strategy on its own, direct policy learning (providing there exists a control policy than can accurately match the joint angle trajectories) is still useful as the first step of a hybrid or goal-directed learning strategy. Providing a good seed or initialization to the learning optimization procedure significantly reduces the learning time, and decreases the chances of the optimization process getting stuck at local minima. Furthermore, in cases where the teacher and student have similar kinematic and dynamic structures, and there is little change in external/environmental condition that relate to the task, direct policy learning alone can be sufficient to learn some simple behaviors.

Finally, two inherent characteristics of this type of research are large amounts of data, and expensive computation. Motion capture systems are capable of providing a great deal of (very accurate) data, which can require considerable time to process even with the currently available computational power. The magnitude of this problem is multiplied by the capture rate, and the complexity of the motion to be captured (a more complex model generally requires more markers). Also, kinematics and dynamics calculations, and non-linear optimization algorithms are computationally expensive by nature, and require considerable processing power as well. Thus efficient, powerful computational and simulation tools are essential for rapid continued progress in this area.

8.2 Improvements to current methods

Several small changes and additions can be made to this work to improve the performance and results of various components, and thus of the overall system. Most of the changes are small, and

can be implemented without the need for changes in other components.

Quaternions for joint angle estimations: Gimbal lock was encountered several times, and though ultimately not significant, it did none the less present a problem in the joint angle estimation component of this work. The fact that gimbal lock was not a major problem was due to a combination of the choice of coordinate system and rotation order, and to the actual motions performed[1]. If the described methods were used to learn other types of motion, gimbal lock could become a major problem. Using quaternions to represent rotation instead of matrices and Euler angle rotations would eliminate gimbal lock. Quaternions are also a more compact representation of rotation, and require less computation that rotation matrices. Join angle estimations then should be implemented using quaternions rather than rotation matrices.

Joint center estimation: The method used for joint center estimation suffer from several drawbacks. First, they require a specific motion (previously called the gym motion) to be preformed by the subjects. Depending on the subject and motion, this may not be practical or even possible. The method is also very sensitive to the actual gym motion preformed, and generally produce unreliable results. The method of Rab et al. [71] avoids these problems and produces reasonable results, but since their method relied on direct measurement of skeletons, it is also not practical as it can't be used for other subject configurations. Therefore, a robust joint center estimation method that does not rely on the subject to perform any special motions is still needed.

Control policy choice: As mentioned previously, the bulk if the error in learning joint trajectories was due to the choice of control policy, namely, the Matsuoka neural oscillator. In addition to the mentioned problems such as topology mismatch, this control policy also suffers in that it essentially requires three parameters to control two aspects of its behavior (recall that α affected the oscillator output amplitude as well as phase, an additional parameter k was necessary to obtain a full 2π phase range, and c controlled final output amplitude). This increases the solution search space and topology complexity, and therefore the time required for the optimization to converge. Though this is not so much a problem during direct policy learning, when only one trajectory is being learned at any one time, it can become a problem during goal oriented learning when all control policies are being modified simultaneously. A new dynamic system for the control policy - one than can be easily configured to more closely match the shape of the oscillations, and that has as few parameters as possible to adjust its behavior - would considerable improve the ability of the system to accurately learn the desired behavior.

Dynamic simulation: As mentioned previously, though desired, a hybrid direct/goal learning strategy was not pursued as the time required to develop a dynamic model for simulation of this task was outside the scope of the work. Dynamic simulation is necessary for goal based learning as it provides an important goal level criteria: drumming impact force. Goal based learning can correct for kinematic and dynamic differences between the teacher and student. This is interesting in the

[1] In most cases, Y component joint angles did not reach $\pm$ 90 °, the point at which gimbal lock occurs

particular task chosen, as drumstick (weight, profile, and length) and drum (size and surface tension) both affect drumming dynamics, and humans are able to compensate for both automatically. Aside from the development of the dynamic simulation itself, including goal-based learning into the described framework would not require much additional work, as only the learning (or, optimization) function would have to be modified to include this criteria. Furthermore, impact forces were recorded for every trial as part of the motion capture process. Developing a dynamic simulation, and performing goal or hybrid based learning would significantly increase learning potential of system.

Goal based or hybrid learning strategies: To further explore the capabilities of imitation learning, more powerful learning strategies than the one used in this work should be implemented and evaluated. Goal based and hybrid learning strategies, including all or a combination of the following optimization criteria in the learning function should be implemented (with weighting) and compared to see which one yields the best results.

- drumstick tip trajectory (position and/or orientation)
- drumstick tip coordinates and/or orientation at impact point
- drumming impact force
- joint angles

Computation and simulation tools: Although Matlab is powerful and significantly facilitates computation and graphical interface development through its matrix manipulation and built in functions, the fact that it's an interpreted language is a major drawback. Specifically, joint angle calculations, forward kinematic calculations, and neural oscillator computations all ran extremely slow, even on a 1.7 GHz Pentium 4 computer with 256 MB of ram. Large amounts of data from motion capture systems, and robot kinematics and dynamics calculations are an inherent part of this research, and thus cannot be avoided. Though it would not affect the results of this work, using different computational tools or development environment to decrease computation and simulation times would be a welcome improvement for the researcher, and could potentially facilitate and speed up the research process.

Extended task analysis: The analysis of the single stroke motions in chapter 5 was a simple analysis, intended to qualitatively highlight visible trends. A more thorough quantitative analysis that models the changes in oscillation parameters (center, frequency, amplitude, and phase) with respect to playing speed and dynamic level would be beneficial. These changes in oscillation properties could then be mapped to control policy parameters. This approach would greatly increase the re-usability of the control policy, as it make it possible to perform one type of task under a large range of conditions with the same (set of) control policy.

8.3 Review of thesis

Chapter 1 of this thesis introduced and gave a synopsis of the thesis and work done, and outlined the approach and scope of the work. Chapter 2 reviewed past and current research undertaken in imitation learning specifically, as well as related aspects including perception and measurement of motion, human body kinematics modeling, extracting kinematic data from optical motion capture data, motion primitives, and dynamic systems as motion primitives. Chapter 3 described the motion capture system, trials, and subjects used in this work, and showed that the system was capable of very accurate spatial measurements. Chapter 4 covered the methods used for extracting usable robot kinematic data from the 3-D motion capture data, and showed that the original measured trajectory can be reproduced with reasonably high accuracy from the estimated joint angles despite poor joint center estimations. Chapter 5 qualitatively and quantitatively analyzed a drumming task, and highlighted visible trends and preferred values for center, frequency, amplitude, and relative phase of oscillation across the different trial conditions and subjects. Chapter 6 examined properties of the Matsuoka neural oscillator, and its applications as a motion primitive, and presented the oscillator configuration used in this work. Chapter 7 presented the imitation learning procedure and results for one type of task, and showed that even with such a simple learning strategy and incompatible motion primitive, the original motion was reproduced with reasonably high accuracy. Appendices A and B contained figures showing the results of the task analysis and direct policy learning respectively.

8.4 Contributions

The contributions of this thesis were as follows:

1. In contrast to previous approaches, which have relied on joint angles sensors to record human rhythmic motion, this work employed an highly accurate optical motion capture system to track key features of interest on the human performer. The advantage this represent over other work is two-fold: First, the employed method is capable of capturing motion with a much higher fidelity; more degrees of freedom can be captured with more accuracy and at a faster sampling rate than previous methods. This translates a more complete, faithful representation, and subsequently to better learning of the behavior. Second, the employed motion capture system expresses motion in 3D Cartesian coordinates (in contrast to joint angle coordinates), which allows for direct comparison between the observed and recreated motion in Cartesian coordinates. This direct comparison is necessary to quantify the accuracy with which a behavior was learned using goal-related variables (such as the end-effector trajectory) instead of simply joint angles.

2. This work attempted to quantify the performance of the described imitation learning model by proposing two suitable metrics (one specifically for rhythmic motion), and presenting the results of applying the metrics to the learned drumming task.

3. This work presented results suggesting the popular Matsuoka oscillator is not a good model of human rhythmic motion, and as a result highlighted an important new characteristic that a primitive for rhythmic motion should possess.

4. This work developed and imitation learning model, and through implementation showed it can be use to teach robots rhythmic tasks involving several degrees of freedom and complex phase requirements between the degrees of freedom.

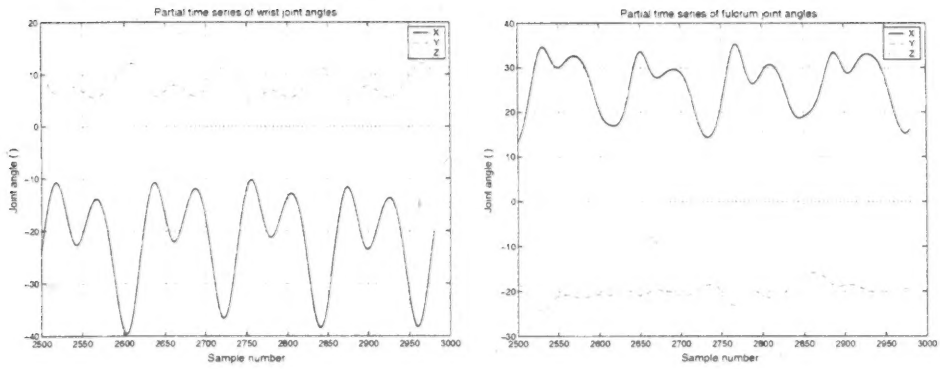
8.5 Further work

Dynamic considerations and fulcrum joint impedance

Since this work focused strictly on the kinematics of movement, one major simplification made in this work is that fulcrum joint angles were actively controlled. This implies that the drumstick position within the hand is actively controlled (presumably) by active control of finger position. In general, this is not the case at all in drumming, and in fact, in most drumming strokes the fingers provide little or no active input to the motion. Furthermore, some drumming strokes played at higher speeds exceed the neuromuscular and motor control bandwidth of humans [13], active finger control is not possible. Instead of active motion then, fingers contribute to drumming by passively modulating the effective impedance of the fulcrum joint [29], and in more complex drumming, a combination of both [38][53]. The action of the fingers can be modeled as a variable rate spring-damper system at the fulcrum joint [29]. This action is also true for many tasks which involve tools. It would be useful to provide robots with ability to learn how to use tools in the same manner as the human teachers do. To this end, the current kinematic model of subject should be augmented to include a dynamic component at the fulcrum joint. A suitable dynamic model of the stick grip should first be developed (a simple mass-damper is a good starting point, which also allows for active control of the rest position), and included in the subject body model. The relevant dynamic parameters (spring and damping coefficients) that lead to the correct desired behavior could then be learned as part of the imitation learning strategy.

Learning of more behaviors

As mentioned previously, due to time constraints, only one of the three captured drumming strokes was analyzed and learned in this work. From preliminary observations, the two strokes that were not learned (double and Moller) produce much more interesting and complex joint angle trajectories. An example is shown in Fig. 8.1. To further examine the effectiveness of all aspects of the proposed imitation learning framework as a robot learning tool, imitation learning of these strokes should be carried out and evaluated.



(a) Wrist.

(b) Fulcrum.

Figure 8.1: Partial time series wrist and fulcrum joint angles for Moller drum stroke, subject three. Two frequency components can clearly be seen here.

Computer vision based motion capture

Though very powerful, commercial motion capture systems can be prohibitively very expensive to many research groups and labs. Also, retro-reflective markers can be intrusive for some task, and still suffer form occlusion and swapped marker trajectories. The capture volume of motion capture systems can also be restrictive for some tasks. Because of these problems, a computer vision based motion measurement tool that uses simple CCD cameras would be advantageous. Much work has been done in this area, [14, 19, 25], but the area is still not well enough developed to be accurate and general enough for most applications [57] (though some researchers [17] are addressing the this). Computer vision based motion capture systems would make this research accessible to more groups, and would make the motion capture process simpler and quicker by eliminating the need for markers. All this could potentially increase the pace of research.

Implementation with a robot

Finally, to further evaluate the described learning framework, and quantify the accuracy of the kinematic and dynamic models, and to make on-line, real-time imitation learning of tasks possible, the learned trajectories should be implemented in a humanoid robot. Furthermore, a system should be in place that interfaces the motion capture system, learning framework, and robot motor control modules.

Metrics for imitation evaluation

The methods described in chapter 7 to evaluate the results of the imitation learning experiment are too simplistic a tool to properly measure how well a motion or behavior has been imitated. There is little work on what featured of movement humans focus on when attempting to imitate a

behavior, or how they evaluate their attempt at imitating it. It is not clear then, what the 'right' features to focus on are for imitation learning. As a metric for measuring imitation performance, the MSE (taken either on joint angle or Cartesian endpoint trajectory) is too strict, and is probably not the feature employed by humans to imitate an observed behavior. Thus, MSE fails to measure the 'right' features for imitation learning. Therefor, more appropriate for evaluating the results of imitation learning experiments are necessary. This is a new area of research, and some methods ands metrics have been proposed by [67] to evaluate this, though they only apply to joint angle comparisons.

Bibliography

[1] Bryan Adams, Cynthia Breazeal, Rodney A. Brooks, and Brian Scassellati. Humanoid robots: A new kind of tool. *IEEE Intelligent Systems and Their Applications: Special Issue on Humanoid Robotics*, 15(4):25–31, August 2000.

[2] Ramesh Amit and Maja J. Matarić. Learning movement sequences from demonstration. In *Proceedings of the International Conference on Development and Learning (ICDL)*, pages 302–306, June 2002.

[3] Ramesh Amit and Maja J. Matarić. Parametric primitives for motor representation and control. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 1, pages 863–868, May 2002.

[4] Motion Analysis. Eagle digital system. http://www.motionanalysis.com/about_mac/eagle system.html, 2002.

[5] Kerstin Dautenhahn Aris Alissandrakis, Chrystopher L. Nehaniv. Do as i do: Correspondences across different robotic embodiments. In *Proceedings of the Fifth German Workshop on Artificial Life (GWAL5)*, pages 143–152, March 2002.

[6] Artur M. Arsenio. Neural oscillator networks for rhythmic control of animats. In *Conference on the Simulation of Adaptive Behavior: From Animals to Animats 6*, August 2000.

[7] Artur M. Arsenio. On stability and error bounds of describing functions for oscillatory control of movements. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2000.

[8] Artur M. Arsenio. Tuning of neural oscillators for the design of rhythmic motions. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1888–1893, 2000.

[9] Christopher G. Atkeson, Joshua G. Hale, Frank Pollick, Marcia. Riley, Shinya Kotosaka, Stefan Schaal, Tomohiro Shibata, Gaurav Tevatia, Ales Ude, Sethu Vijayakumar, and Mitsuo Kawato. Using humanoid robots to study human behavior. *IEEE Intelligent Systems and Their Applications: Special Issue on Humanoid Robotics*, 15(4):46–56, August 2000.

[10] Aude Billard. Learning motor skills by imitation: a biologically inspired robotic model. *Cybernetics and Systems*, 32(1-2):155–193, 2001.

[11] Aude Billard and Maja J. Matarić. "betty: Robot, play with me! robot: O.k. how do we play? betty: You watch me and do like i do. look!". In *Proceedings, Workshop on Interactive Robotics and Entertainment (WIRE-2000)*, April 2000.

[12] Aude Billard and Maja J. Matarić. Learning human arm movements by imitation: Evaluation of a biologically-inspired connectionist architecture. *Robotics & Autonomous Systems*, 37(2):1–16, November 2001.

[13] Kenneth R. Boff and Janet E. Lincoln. *Engineering Data Compendium: Human Perception and Performance*. Wright-Patterson A.F.B., Ohio : Harry G. Armstrong Aerospace Medical Research Laboratory, 1988.

[14] Andrea Bottino and Aldo Laurentini. A silhouette based technique for the reconstruction of human movement. *Computer Vision and Image Understanding*, 83(1):79–95, July 2001.

[15] Ronan Boulic, Pascal Fua, Lorna Herda, Marius Silaghi, Jean S. Monzani, Luciana Nedel, and Daniel Thalmann. An anatomic human body for motion capture. In *European Multimedia, Embedded Systems and Electronic Commerce Conference*, September 1998.

[16] Kwang-Jin Choi, Sang-Hyun Park, and Hyeong-Seok Ko. Processing motion capture data to achieve positional accuracy. *Graphical Models and Image Processing*, 61(5):260–273, September 1999.

[17] Chi-Wei Chu, Odest Chadwicke Jenkins, and Maja J. Matarić. Markerless kinematic model and motion capture from volume sequences. In *To appear in IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2003*. IEEE, 2003.

[18] P.I. Corke. A robotics toolbox for MATLAB. *IEEE Robotics and Automation Magazine*, 3(1):24–32, March 1996.

[19] Nicola D'Apuzzo. Surface measurement and tracking of human body parts from multi-image video sequences. *ISPRS Journal of Photogrammetry and Remote Sensing*, 56(5-6):360–375, August 2002.

[20] K. Anders Ericsson and Neil Charness. Expert performance its structure and acquisition. *American Psychologist*, 49(8):725–747, August 1994.

[21] K. Anders Ericsson, Ralf T. Krampe, and Clemens Tesch-Römer. The role of deliberate practice in the acquisition of expert performance. *Psychological Review*, 100(3):363–406, July 1993.

[22] Pascal Fua, Armin Gruen, Ralf Plaänkers, Nicola D'apuzzo, and Daniel Thalmann. Human body modeling and motion analysis from video sequences. In *International Symposium on Real Time Imaging and Dynamic Analysis*, June 1998.

[23] Pascal Fua, Lorna Herda, Ralf Plänkers, and Ronan Boulic. Human shape and motion recovery using animation models. In *19th Congress of International Society for Photogrammetry and Remote Sensing*, July 2000.

[24] Pascal Fua, Ralf Plänkers, and Daniel Thalmann. Realistic human body modeling. In *Fifth International Symposium on the 3-D Analysis of Human Movement*, June 1998.

[25] Pascal Fua, Ralf Plänkers, and Daniel Thalmann. From synthesis to analysis: Fitting human animation models to image data. In *Computer Graphics International*, pages 4–11, June 1999.

[26] Simon F. Giszter, Karen A. Moxon, Ilya A. Rybak, and John K. Chapin. A neurobiological perspective on humanoid robot design. *IEEE Intelligent Systems and Their Applications: Special Issue on Humanoid Robotics*, 15(4):64–69, August 2000.

[27] Stephen Grossberg, Christopher Pribe, and Michael A. Cohen. Neural control of inter-limb oscillations i. human bi-manual coordination. *Biological Cybernetics*, 77(2):131–140, May 1997.

[28] Yves Guiard. Fitts' law in the discrete vs. cyclical paradigm. *Human Movement Science*, 16(1):97–131, February 1997.

[29] Aram Z. Hajian, Daniel S. Sanchez, and Robert D. Howe. Drum roll: Increasing bandwidth through passive impedance modulation. In *IEEE International Conference on Robotics and Automation*, pages 2294–2299, April 1997.

[30] Joshua G. Hale and Frank E. Pollick. Playing 'sticky hands' with a humanoid robot. In *First IEEE-RAS International Conference on Humanoid Robots (Humanoids 2000)*. MIT, September 2000.

[31] Lorna Herda, Pascal Fua, Ralf Plänkers, Ronan Boulic, and Daniel Thalmann. Using skeleton-based tracking to increase the reliability of optical motion capture. In *Human Movement Science*, volume 20, pages 313–341, June 2001.

[32] Auke Jan Ijspeert, Jun Nakanishi, and Stefan Schaal. Trajectory formation for imitation with nonlinear dynamical systems. In *IEEE International Conference on Intelligent Robots and Systems (IROS 2001)*, 2001.

[33] Auke Jan Ijspeert, Jun Nakanishi, and Stefan Schaal. Learning rhythmic movements by demonstration using nonlinear oscillators. In *IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 958–963, 2002.

[34] Auke Jan Ijspeert, Jun Nakanishi, and Stefan Schaal. Movement imitation with nonlinear dynamical systems in humanoid robots. In *International Conference on Robotics and Automation (ICRA2002)*, 2002.

[35] Skill Technologies inc. 6DR polhemus systems. http://www.skilltechnologies.com/6DR-Polhemus.htm, 2001.

[36] Hirokazu Iwase and Atsuo Murata. Chaotic features of rhythmic joint movement. In *IEICE Transactions on Information and Systems*, volume E85-D, pages 1175–1179, July 2002 July 2002.

[37] Odest Chadwicke Jenkins, Maja J. Matarić, and Stefan Weber. Primitive-based movement classification for humanoid imitation. In *First IEEE-RAS International Conference on Humanoid Robotics (Humanoids-2000)*, September 2000.

[38] Brian Jones. Informal conversations on drumming technique, July 31 2002.

[39] Murali P. Kadaba and Rebecca Stine. Real-time movement analysis: Techniques for the new millennium. In *5th Annual Gait and Clinical Movement Analysis Meeting*, April 2000.

[40] Neri Kafkafi and Ilan Golani. A traveling wave of lateral movements coordinates both turning and forward walking in the ferret. *Biological Cybernetics*, 78(6):441–435, 1998.

[41] B.A. Kay, E.L. Saltzman, J.A.S Kelso, and G. Schöner. The space-time behavior of single and bimanual rhythmical movements: Data and a limit cycle model. *Journal of Experimental Psychology: Human Perception and Performance*, 13:178–192, 1987.

[42] Petra Köpf-Maier. *Wolf-Heidegger's Atlas of Human Anatomy, 5^{th} edition*. Karger, 2000.

[43] Shinya Kotosaka and Stefan Schaal. Synchronized robot drumming by neural oscillator. In *The International Symposium on Adaptive Motion of Animals and Machines*, 2000.

[44] M.A. Lafortune, P.R. Cavanaugh, H.J. Sommer, and A. Kalenka. Three-dimensional kinematics of the human knee during walking. *Journal of Biomechanics*, 25(4):347–357, April 1992.

[45] Suzanne LaScalza, Jane Arico, and Richard Hughes. Effect of metal and sampling rate on accuracy of flock of birds electromagnetic tracking system. *Journal of Biomechanics*, 35:141 144, 2003.

[46] Daeyeol Lee. Learning of spatial and temporal patterns in sequential hand movements. *Cognitive Brain Research*, 9:35–39, 2000.

[47] Maja J. Matarić. Getting humanoids to move and imitate. *IEEE Intelligent Systems and Their Applications: Special Issue on Humanoid Robotics*, 15(4):18–24, August 2000.

[48] Maja J. Matarić. Sensory-motor primitives as a basis for learning by imitation: Linking perception to action and biology to robotics. In Kerstin Dautenhahn and Chrystopher Nehaniv, editors, *Imitation in Animals and Artifacts, Kerstin Dautenhahn and Chrystopher Nehaniv*, pages 392–422. MIT Press, 2002.

[49] Maja J. Matarić and Marc Pomplun. Fixation behavior in observation and imitation of human movement. *Cognitive Brain Research*, 7(2):191–202, October 1998.

[50] The MathWorks. How can i use fft to obtain simple spectral analysis plots? tech-note 1702. Technical report, The MathWorks, March 2003.

[51] Kiyotoshi Matsuoka. Sustained oscillations generated by mutually inhibiting neurons with adaptation. In *Biological Cybernetics*, volume 52, pages 367–376, 1985.

[52] Kiyotoshi Matsuoka. Mechanisms of frequency and pattern control in the neural rhythm generators. In *Biological Cybernetics*, volume 56, pages 345–353, 1987.

[53] John McCormick. Informal conversations on drumming technique, July 17 2002.

[54] A. D. Milne, D. G. Chess, J. A. Johnson, and G. J. W. King. Accuracy of an electromagnetic tracking device: a study of the optimal operating range and metal interference. *Journal of Biomechanics*, 29(6):791–793, June 1996.

[55] Hiroyuki Miyamoto and Mitsuo Kawato. A tennis serve and upswing learning robot based on bi-directional theory. *Neural Networks*, 11(7-8):1331–1344, October 1998.

[56] Hiroyuki Miyamoto, Stefan Schaal, Francesca Gandolfo, Hiroaki Gomi, Yasuharu Koike, Rieko Osu, Eri Nakano, Yasuhiro Wada, and Mitsuo Kawato. A kendama learning robot based on bi-directional theory. *Neural Networks*, 8(9):1281–1302, November 1996.

[57] Thomas B. Moeslund and Erik Granum. A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding*, 81(3):231–268, March 2001.

[58] Jean S. Monzani, P. Baerlocher, Ronan Boulic, and Daniel Thalmann. Using an intermediate skeleton and inverse kinematics for motion retargeting. In *European Association for Computer Graphics*, August 2000.

[59] Meta Motion. Gypsy 3 motion capture system. http://www.metamotion.com/gypsy/gypsy-motion-capture-system.htm, 2000.

[60] K.M. Newell, L.G. Carlton, S. Kim, and C.H. Chung. Space-time accuracy of rapid movements. *Journal of Motor Behavior*, 25(1):8–20, 1993.

[61] University of Southern California. Demonstrations of articulated control. http://www-robotics.usc.edu/ agents/Links/Research/macarena.html, 1999.

[62] University of Southern California. Imitation test-beds. http://www-robotics.usc.edu/ agents/Links/Research/imit-testbeds.html, 2000.

[63] University of Southern California. Humanoid dynamic simulator. http://robotics.usc.edu/ agents/simulator.html, 2001.

[64] Massachusetts Institute of Technology. Mit artificial intelligence laboratory: Humanoid robotics group. http://www.ai.mit.edu/projects/humanoid-robotics-group/.

[65] Charles L. Phillips and Royce D. Harbor. *Feedback Control Systems, 3^{rd} edition*. Prentice Hall, 1996.

[66] Nancy S. Pollard, Jessica K. Hodgins, Marcia J. Riley, and Christopher G. Atkeson. Adapting human motion for the control of a humanoid robot. In *IEEE Conference on Robotics and Automation*, May 2002.

[67] Marc Pomplun and Maja J. Matarić. Technical report iris-02-410: A segmentation algorithm for the comparison of human limb trajectories. Technical report, Institute for Robotics and Intelligent Systems, School of Engineering, University of Southern California, December 2001.

[68] François Poulin and L.-P. Amiot. Interference during the use of an electromagnetic tracking system under or conditions. *Journal of Biomechanics*, 35:733–737, 2002.

[69] Christopher Pribe, Stephen Grossberg, and Michael A. Cohen. Neural control of inter-limb oscillations ii. biped and quadruped gaits and bifurcations. *Biological Cybernetics*, 77(2):141–152, May 1997.

[70] R.A. Propenko, A.A. Frolov, E.V. Biryukova, and A. Roby-Brami. Assessment of the accuracy of a human arm model with seven degrees of freedom. *Journal of Biomechanics*, 34:177–185, 2001.

[71] George Rab, Kyria Petuskey, and Anita Bagley. A method for determination of upper extremity kinematics. *Gait and Posture*, 15(2):113–119, April 2002.

[72] James G. Richards. The measurement of human motion: A comparison of commercially available systems. *Human Movement Science*, 18(5):589–602, October 1999.

[73] Marcia Riley, Ales Ude, and Christopher G. Atkeson. Methods for motion generation and interaction with a humanoid robot: Case studies of dancing and catching. In *AAAI and CMU Workshop on Interactive Robotics and Entertainment*, April 2000.

[74] Sarcos. Sarcos sensuit. http://www.sarcos.com/humanintfc.html.

[75] SARCOS. Sarcos. http://www.sarcos.com/, 2003.

[76] Stefan Schaal. Is imitation learning the route to humanoid robots? *Trends in Cognitive Sciences*, 3(6):233–242, 1999.

[77] Stefan Schaal, Auke Jan Ijspeert, and Aude Billard. Computational approaches to motor learning by imitation. *Philosophical Transaction of the Royal Society of London: Series B, Biological Sciences*, 358(1431):537–547, 2003.

[78] Stefan Schaal, Shinya Kotosaka, and Dagmar Sternad. Nonlinear dynamic systems as movement primitives. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids 2000)*, Cambridge, MA, 2000.

[79] Stefan Schaal and Dagmar Sternad. Programmable pattern generators. In *International Conference on Computational Intelligence in Neuroscience*, pages 44–51, Research Triangle Park, NC, Oct.24-28 1998.

[80] Japan Science and Technology Corporation. Db's home page. http://www.kawato.jst.go.jp/DB/, 2000.

[81] Japan Science and Technology Corporation. Kawato dynamic brain project. http://www.kawato.jst.go.jp/index.html, 2003.

[82] Marius-Calin Silaghi, Ralf Plänkers, Ronan Boulic, Pascal Fua, and Daniel Thalmann. Local and global skeleton fitting techniques for optical motion capture. In *International Workshop on Modeling and Motion Capture Techniques for Virtual Environments*, pages 26–40, November 1998.

[83] Dagmar Sternad, William J. Dean, and Karl M. Newell. Force and timing variability in rhythmic uni-manual tapping. *Journal of Motor Behavior*, 32(3):249–267, 2000.

[84] Dagmar Sternad, M. T. Turvey, and Elliot Saltzman. Dynamics of 1:2 coordination: Generalizing relative phase. *Journal of Motor Behavior*, 31(3):207–223, 1999.

[85] Ales Ude, Curtis Man, Marcia Riley, and Christopher G. Atkeson. Automatic generation of kinematic models for the conversion of human motion capture data into humanoid robot motion. In *IEEE-RAS International Conference on Humanoid Robots*, September 2000.

[86] Matthew M. Williamson. Exploiting natural dynamics in robot control. In *Proceedings Fourth European Meeting on Cybernetics and Systems Research (EMCSR '98)*, 1998.

[87] Matthew M. Williamson. Neural control of rhythmic arm movements. In *Neural Networks Special Issue on Neural Control of Movement*, volume 11, pages 1379–1394, October 1998.

[88] Matthew M. Williamson. Designing rhythmic motions using neural oscillators. In *IEEE International Conference on Intelligent Robots and Systems*, volume 1, pages 494–500, 1999.

[89] Matthew M. Williamson. *Robot Arm Control Exploiting Natural Dynamics*. PhD thesis, Massachusetts Institute of Technology, June 1999.

[90] Ge Wu, Sorin Siegler, Paul Allarda, Chris Kirtleya, Alberto Leardini, Dieter Rosenbaum, Mike Whittle, Darryl D. D'Lima, Luca Cristofolini, and Hartmut Witte. Isb recommendation on definitions of joint coordinate system of various joints for the reporting of human joint motion–part i: ankle, hip, and spine. *Journal of Biomechanics*, 35(4):543–548, April 2002.

[91] Victor B. Zordan and Jessica K. Hodgins. Tracking and modifying upper-body human motion data with dynamic simulation. In *Computer Animation and Simulation '99, Eurographics Animation Workshop*, pages 13–22, September 1999.

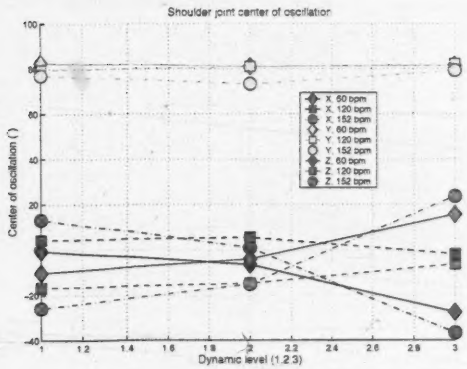
Appendix A

Analysis Results of Single Stroke Drumming Task

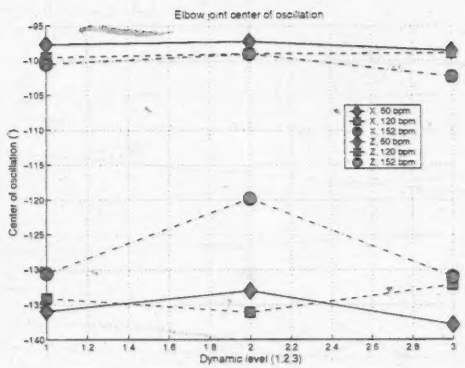
Following, the results of center, frequency and amplitude of oscillation, and relative phase analysis of the four involved segments and all relevant degrees of freedom are presented for each subject. The analysis attempted to highlight any obvious trends or preferred values in each category with respect to task frequency or dynamic level. Results are plotted vs. dynamic level (1=soft, 2=moderate, 3=loud) for each of the task frequencies (60 bpm=diamond, 120 bpm=square, highest task frequency=circle). Below each figure, any visible trends are summarized. Some notes on the presentation should be made: For points where no oscillation occurred, relative phase angles were assigned a value of zero. Also, when component showed preferred values, the mean of all points excluding those where no oscillation occurred are presented.

A.1 Subject one results

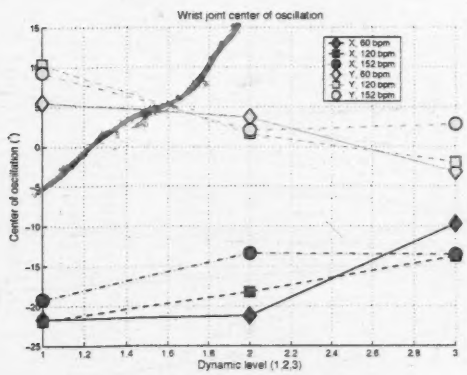
Center of oscillation



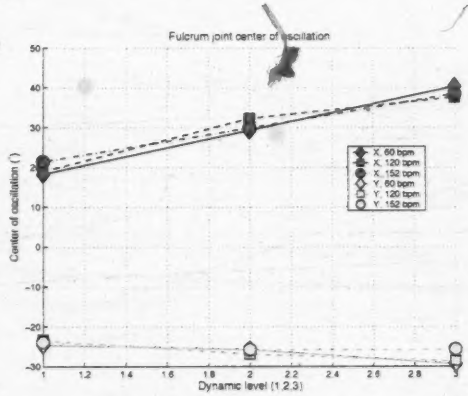
(a) Shoulder joint center of oscillation.



(b) Elbow joint center of oscillation



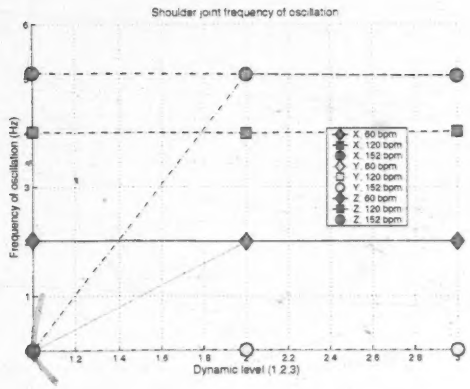
(c) Wrist joint center of oscillation



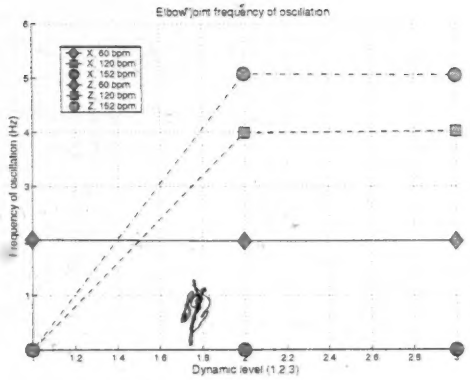
(d) Fulcrum joint center of oscillation

Figure A.1: Results of center of oscillation analysis for the single stroke task, subject one. The center of oscillation for the shoulder X and Z component showed some variability, but no apparent trend was visible. The Y component was constant for all conditions. Mean and standard deviation values for the X, Y, and Z components were -6° and 16°, 80° and 3° and -6° and 16°respectively. The center of oscillations for the elbow X and Z component of the elbow were relatively constant for all conditions. Mean and standard deviation values for the X and Z components were -99° and 2°, and -132° and 5° respectively. The center of oscillation for the wrist X and Y components showed a slight increase and decrease respectively as dynamic level increased. Values ranged from approximately -22° to -10° for the X component, and 10° to -3° for the Y component. The center of oscillation for the fulcrum X and Y components showed a slight increase and decrease respectively with increasing dynamic level. Values ranged from approximately 18° to 40° for the X component, and -23° to -29° for the Y component.

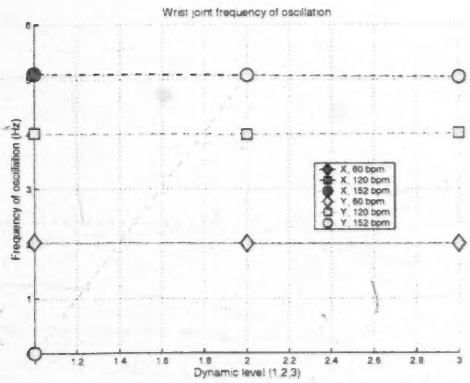
Frequency of oscillation



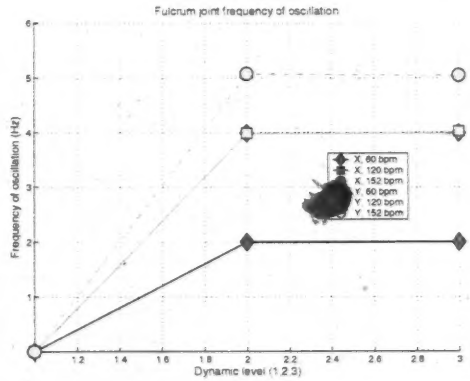
(a) Shoulder joint frequency of oscillation



(b) Elbow joint frequency of oscillation



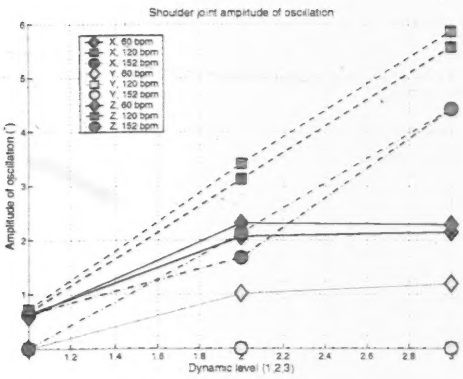
(c) Wrist joint frequency of oscillation



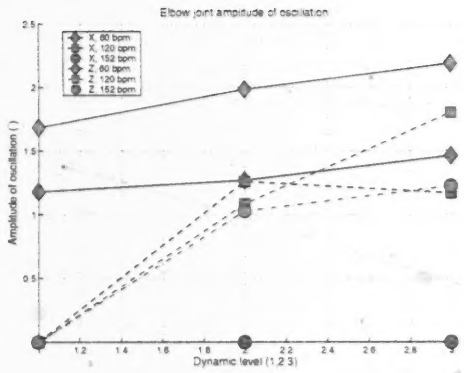
(d) Fulcrum joint frequency of oscillation

Figure A.2: Results of frequency of oscillation analysis for the single stroke task, subject one. With one exception the oscillating frequency of the shoulder X and Z components matched the playing speed for all conditions. The Y component oscillated only at low a low task frequency, and also followed the playing speed. The elbow Z component oscillated at the task frequency for all but two conditions. The X component did not oscillate at low dynamic levels or at high tasks speeds, but otherwise followed the task frequency. With one exception when no oscillation occurred, the oscillating frequency of both the X and Y components of the wrist joint followed the playing frequency for all conditions. At a low dynamic level, no oscillations occurred in either fulcrum component. At a low task frequency, the oscillating frequency of Y component was twice the task frequency. Under all other conditions, the oscillating frequency of both components followed the task frequency.

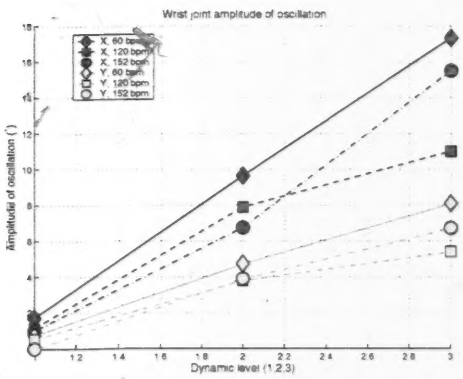
Amplitude of oscillation



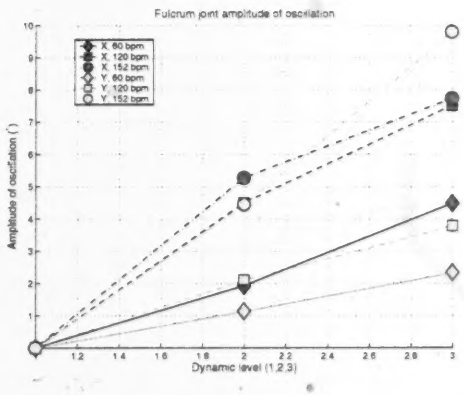
(a) Shoulder joint amplitude of oscillation



(b) Elbow joint amplitude of oscillation



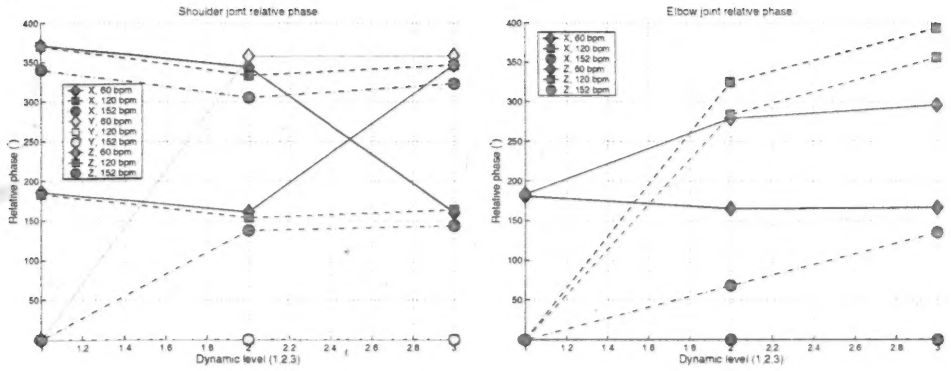
(c) Wrist joint amplitude of oscillation



(d) Fulcrum joint amplitude of oscillation

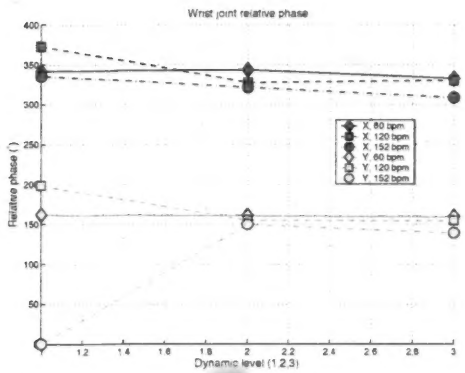
Figure A.3: Results of amplitude of oscillation analysis for the single stroke task, subject one. The amplitude of oscillation of the shoulder X and Z components showed and increase with increasing dynamic level. Amplitudes ranged from 1° to 6° for both components. The Y component also followed this trend, but to a smaller degree. Amplitudes ranged from 0° to approximately 1.5°. The Oscillating amplitudes of the elbow X and Z components show a slight increase with dynamic level, and a slight decrease respect to task speed. As with the shoulder, the oscillations were small, in the range of 0° to 2°. The amplitude of oscillation of both the X and Y wrist components showed a general increase with increasing dynamic level. Values ranged from 1° to 17° for the X component, and 0° to 9° for the Y component. The amplitude of oscillation of both the X and Y fulcrum components showed a general increase with increasing dynamic level and task frequency. Values ranged from 0° to 8° for the X component, and 0° to 10° for the Y component.

Relative phase

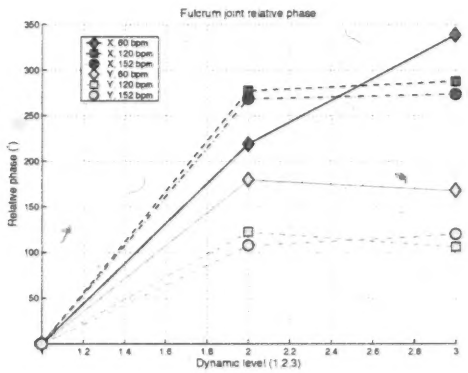


(a) Shoulder joint relative phase

(b) Elbow joint relative phase



(c) Wrist joint relative phase

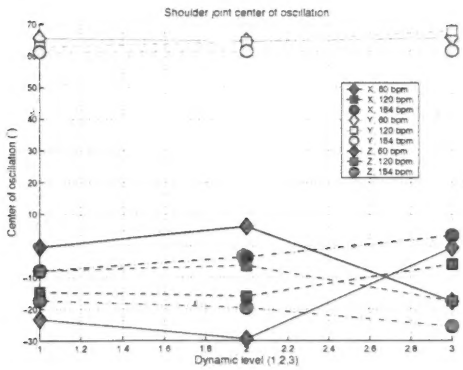


(d) Fulcrum joint relative phase

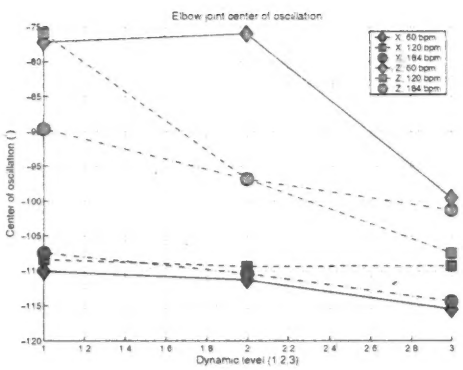
Figure A.4: Results of relative phase analysis for the single stroke task, subject one. Excluding conditions where no oscillation occurred, the shoulder X component showed a preferred value of 321°. Similarly, excluding conditions of 60 bpm task speed at a high dynamic level, and 152 bpm task speed at a low dynamic level, the shoulder Z component showed a preferred value of 184°. Additionally, the phase values of these two components were always 180°apart, or anti-phase. Under the few conditions for which oscillation occurred, the shoulder Y component showed a preferred value of 357°. The elbow X and Z components showed no preferred values under any conditions. Phase angles for both the X and Y components of the wrist showed strong preferred values of 335° and 160° respectively. Excluding low task dynamic levels where no oscillation occurred, phase angles for the the X and Y components of the fulcrum also showed preferred values of 277° and 134 respectively.

A.2 Subject two results

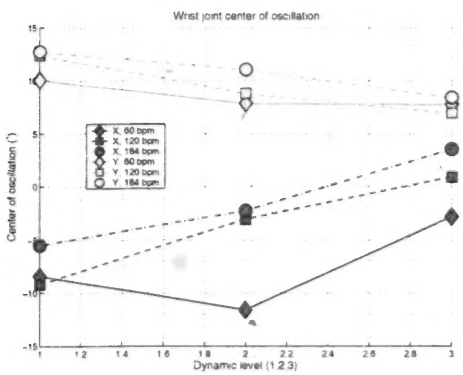
Center of oscillation



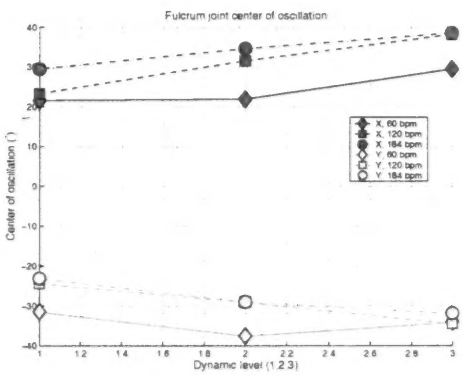
(a) Shoulder joint center of oscillation.



(b) Elbow joint center of oscillation



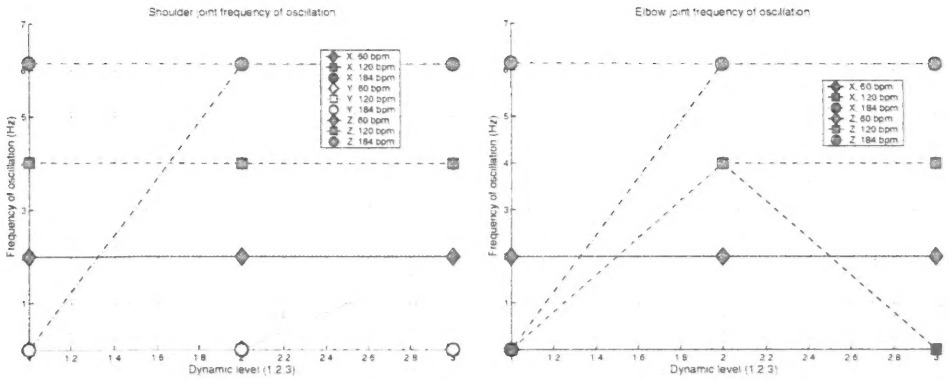
(c) Wrist joint center of oscillation



(d) Fulcrum joint center of oscillation

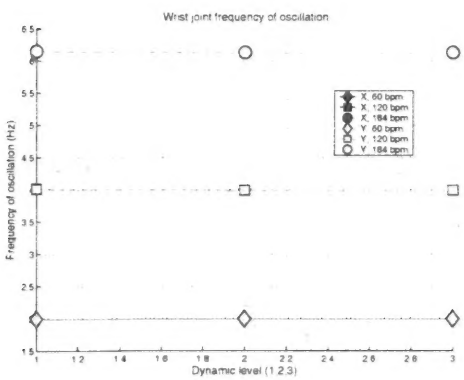
Figure A.5: Results of center of oscillation analysis for the single stroke task, subject two. The center of oscillation for the shoulder X and Z component showed some variability, but no apparent trend was visible. The Y component was constant for all conditions. Mean and standard deviation values for the X, Y, and Z components were -6° and 8°, 63° and 2°, and -16° and 10°respectively. The center of oscillation for the elbow X component was relatively constant for all conditions. Mean and standard deviation values for -111° and 3°. The Z component however, showed a slight decrease (indicating medial rotation) of about 30° with increasing dynamic level. Values ranged from -76° to -107° The center of oscillation for the wrist X and Y components showed a slight increase and decrease respectively as dynamic level increased. Values ranged from approximately -9° to 4° for the X component, and 13° to 7° for the Y component. The center of oscillation for the fulcrum X and Y components showed a slight increase and decrease respectively with increasing dynamic level. Values ranged from approximately 22° to 38° for the X component, and -23° to -38° for the Y component.

Frequency of oscillation

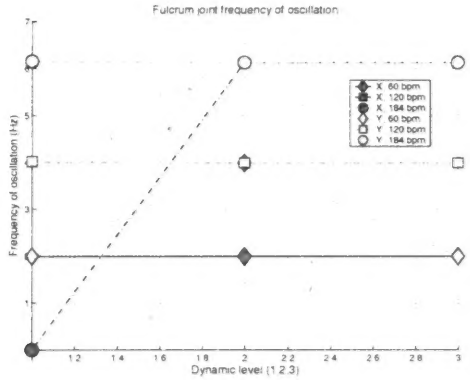


(a) Shoulder joint frequency of oscillation

(b) Elbow joint frequency of oscillation



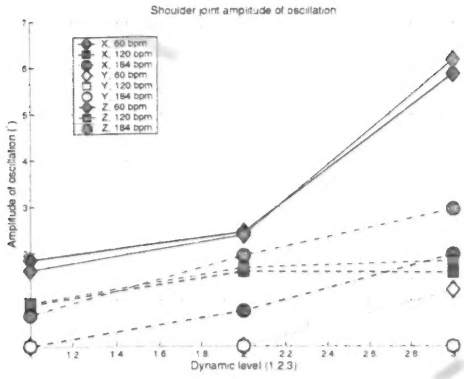
(c) Wrist joint frequency of oscillation



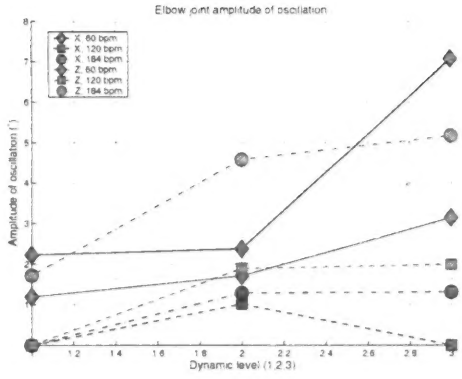
(d) Fulcrum joint frequency of oscillation

Figure A.6: Results of frequency of oscillation analysis for the single stroke task, subject two. With one exception the oscillating frequency of the shoulder X and Z components matched the playing speed for all conditions. The Y component generally did not oscillate. The elbow Z component oscillated at the task frequency for all but one condition. The X component did not oscillate under some conditions, but otherwise followed the task frequency. The oscillating frequency of both the X and Y components of the wrist joint followed the playing frequency for all conditions. With one exception, the fulcrum X and Y components oscillated at task frequency.

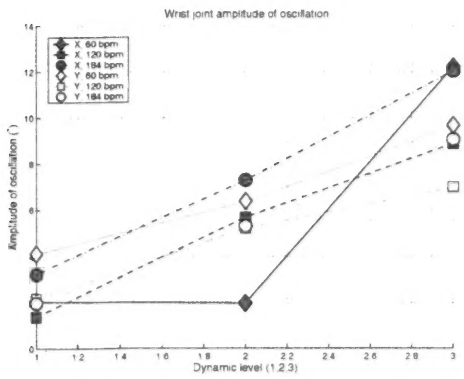
Amplitude of oscillation



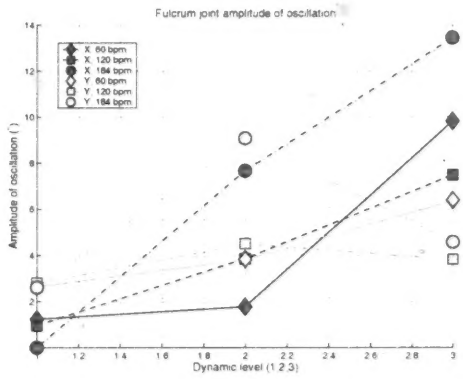
(a) Shoulder joint amplitude of oscillation



(b) Elbow joint amplitude of oscillation



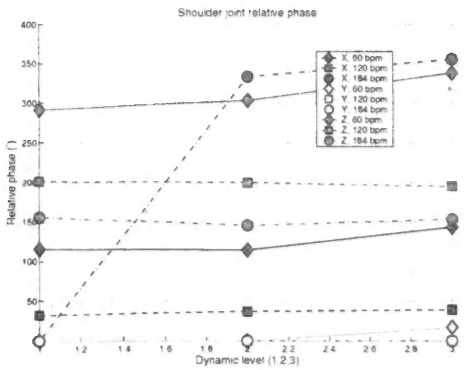
(c) Wrist joint amplitude of oscillation



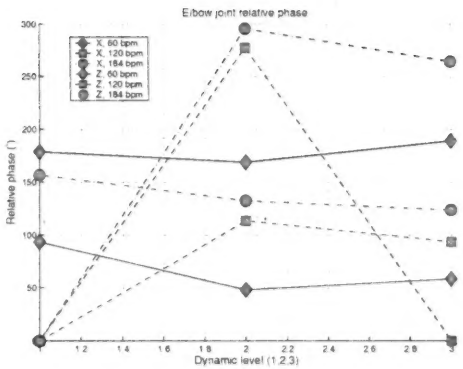
(d) Fulcrum joint amplitude of oscillation

Figure A.7: Results of amplitude of oscillation analysis for the single stroke task, subject two. The amplitude of oscillation of the shoulder X and Z components showed and increase with increasing dynamic level. Amplitudes ranged from 0° to 6° for both components. The Y in general did not oscillate. The Oscillating amplitudes of the elbow X and Z components show a slight increase with dynamic level, and no trend with respect to task speed. Values were in the range of 0° to 7° and 0° to 5° for the X and Z components respectively. The amplitude of oscillation of both the X and Y wrist components showed a general increase with increasing dynamic level. Values ranged from 1° to 12° for the X component, and 2° to 10° for the Y component. The amplitude of oscillation of both the X and Y fulcrum components showed a general increase with increasing dynamic level. Values ranged from 0° to 13° for the X component, and 3° to 9° for the Y component.

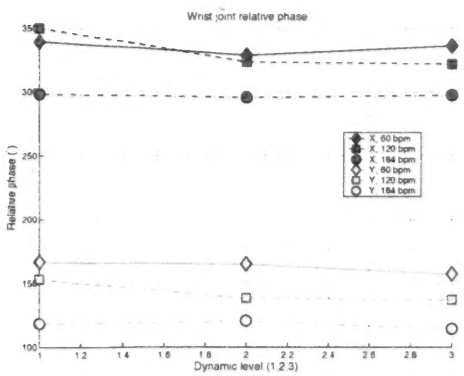
Relative phase



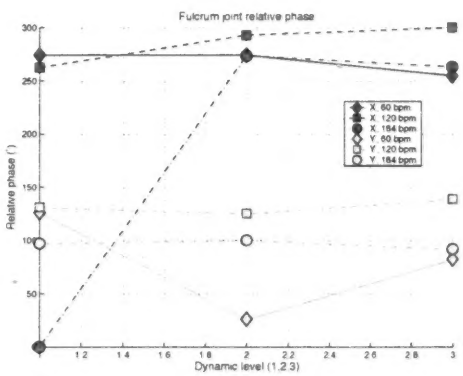
(a) Shoulder joint relative phase



(b) Elbow joint relative phase



(c) Wrist joint relative phase

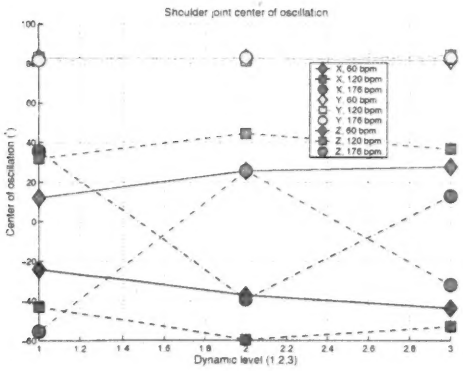


(d) Fulcrum joint relative phase

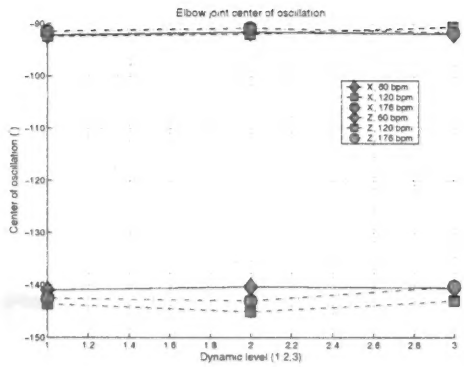
Figure A.8: Results of relative phase analysis for the single stroke task, subject two. No apparent trend wast visible for any component of both the shoulder and elbow joints. However, the phase values of the shoulder X and Z components were always 180° apart, or anti-phase. Phase angles for both the X and Y components of the wrist showed strong preferred values of 321° and 141°respectively. Excluding points of no oscillation, phase angles for the the X and Y components of the fulcrum also showed preferred values of 274° and 102 respectively.

A.3 Subject three results

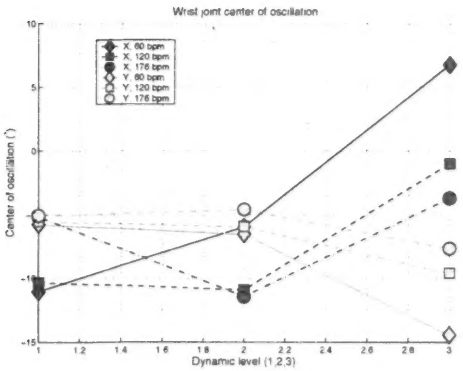
Center of oscillation



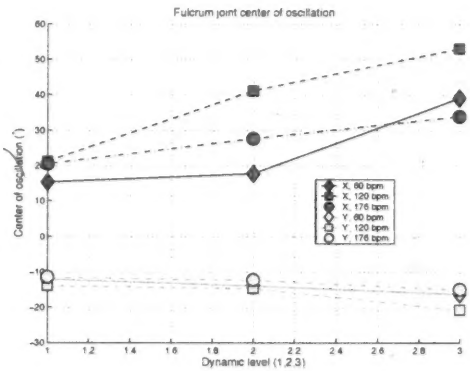
(a) Shoulder joint center of oscillation.



(b) Elbow joint center of oscillation



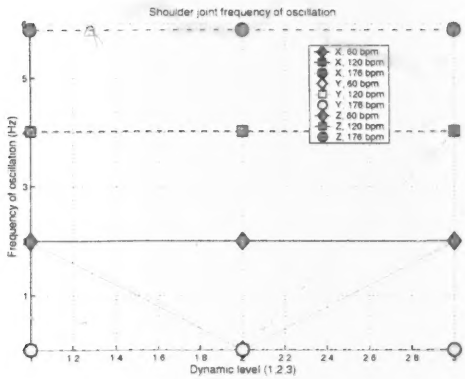
(c) Wrist joint center of oscillation



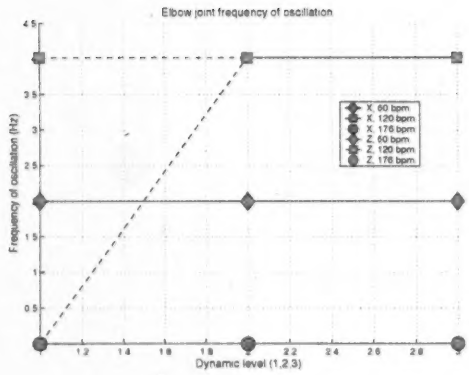
(d) Fulcrum joint center of oscillation

Figure A.9: Results of center of oscillation analysis for the single stroke task, subject three. The center of oscillation for the shoulder X and Z component showed some variability, but no apparent trend was visible. The Y component was constant for all conditions. Mean and standard deviation values for the X, Y, and Z components were -43° and 11°, 83° and 1°, and 29° and 10°respectively. The center of oscillations for the elbow X and Z component of the elbow were relatively constant for all conditions. Mean and standard deviation values for the X and Z components were -92° and 1°, and -142° and 2° respectively. The center of oscillation for the wrist X and Y components showed a slight increase and decrease respectively as dynamic level increased. Values ranged from approximately -11° to 7° for the X component, and -5° to -14° for the Y component. The center of oscillation for the fulcrum X and Y components showed a slight increase and decrease respectively with increasing dynamic level. Values ranged from approximately 15° to 53° for the X component, and -12° to -21° for the Y component.

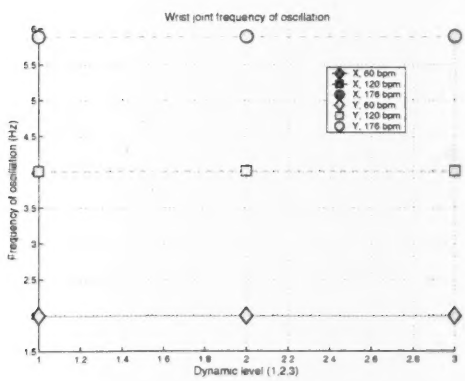
Frequency of oscillation



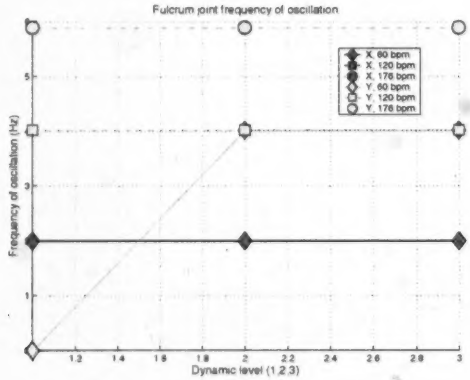
(a) Shoulder joint frequency of oscillation



(b) Elbow joint frequency of oscillation



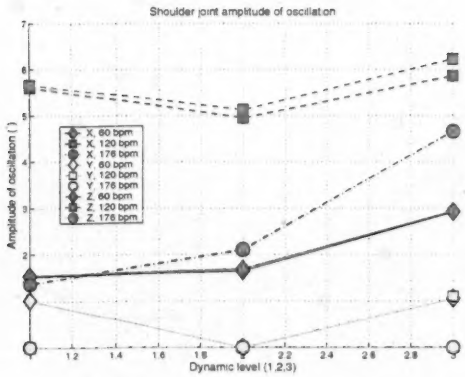
(c) Wrist joint frequency of oscillation



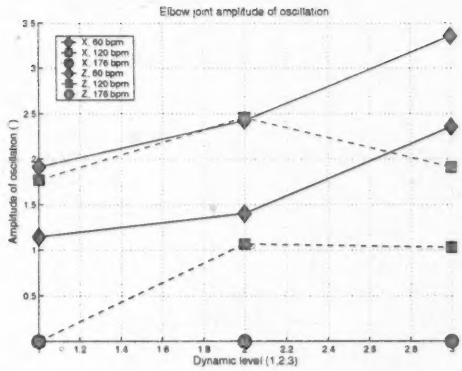
(d) Fulcrum joint frequency of oscillation

Figure A.10: Results of frequency of oscillation analysis for the single stroke task, subject three. The oscillating frequency of the shoulder X and Z components matched the playing speed for all conditions. The Y component generally did not oscillate consistently. The elbow X and Z components oscillated at the task frequency for all but one condition. The oscillating frequency of both the X and Y components of the wrist joint followed the playing frequency for all conditions. With one exception, the fulcrum X and Y components oscillated at task frequency.

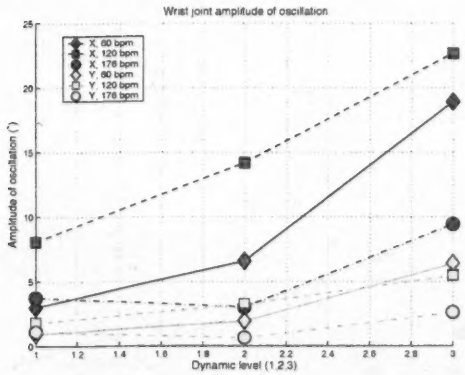
Amplitude of oscillation



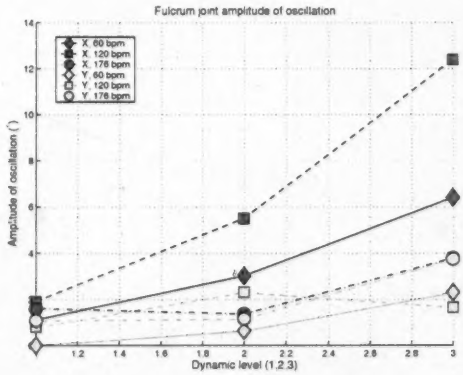
(a) Shoulder joint amplitude of oscillation



(b) Elbow joint amplitude of oscillation



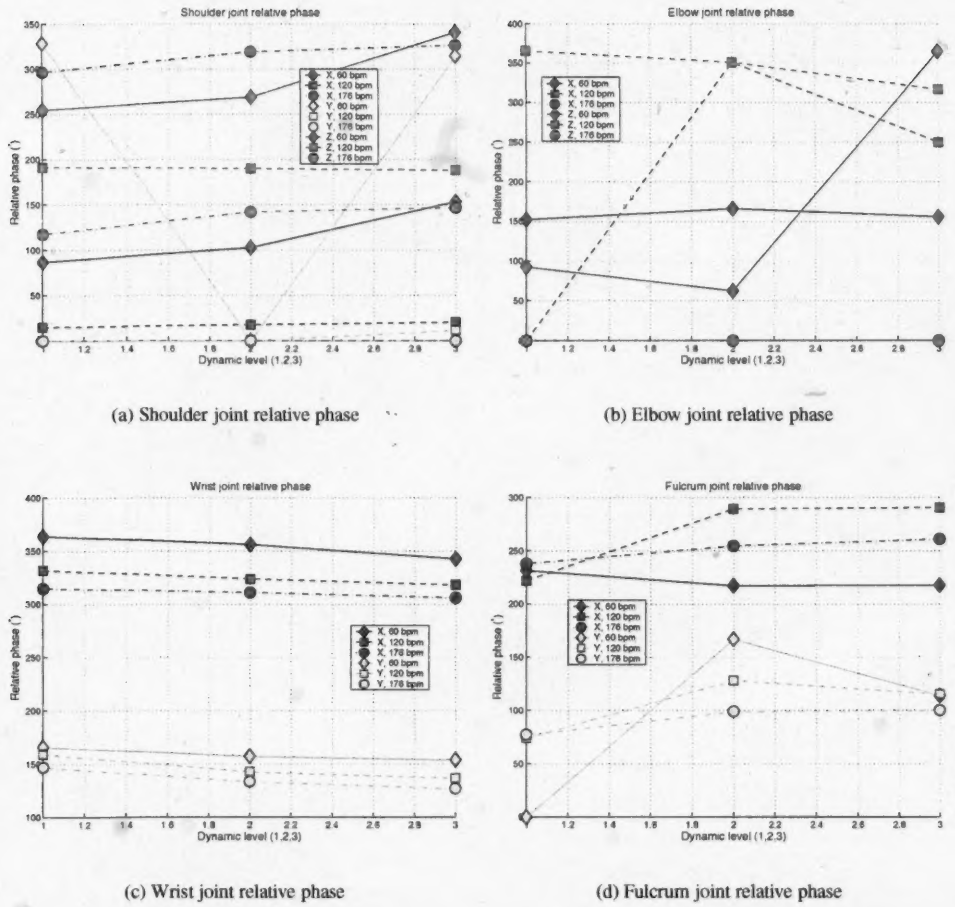
(c) Wrist joint amplitude of oscillation



(d) Fulcrum joint amplitude of oscillation

Figure A.11: Results of amplitude of oscillation analysis for the single stroke task, subject three. The amplitude of oscillation of the shoulder X and Z components showed and increase with increasing dynamic level. Amplitudes ranged from 1° to 6° for both components. The Y in general did not oscillate. The Oscillating amplitudes of the elbow X and Z components show a slight increase with dynamic level, and a slight decrease with respect to task speed. Values were in the range of 0° to 2° and 0° to 3° for the X and Z components respectively. The amplitude of oscillation of both the X and Y wrist components showed a general increase with increasing dynamic level. Values ranged from 3° to 23° for the X component, and 1° to 6° for the Y component. The amplitude of oscillation of both the X and Y fulcrum components showed a general increase with increasing dynamic level. Values ranged from 1° to 12° for the X component, and 0° to 4° for the Y component.

Relative phase



(a) Shoulder joint relative phase

(b) Elbow joint relative phase

(c) Wrist joint relative phase

(d) Fulcrum joint relative phase

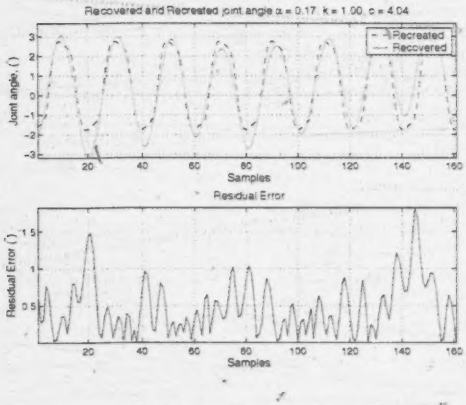
Figure A.12: Results of relative phase analysis for the single stroke task, subject three. No apparent trend wast visible for any component of both the shoulder and elbow joints. However, the phase values of the shoulder X and Z components were again always 180° apart, or anti-phase. Phase angles for both the X and Y components of the wrist showed strong preferred values of 329° and 147°respectively. Excluding point of no oscillation, phase angles for the the X and Y components of the fulcrum also showed preferred values of 274° and 109 respectively.

Appendix B

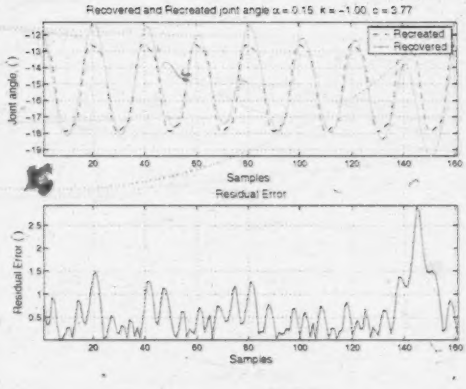
Learning Results

Following, the results of direct policy learning of the demonstrated joint trajectories are presented for the single stroke drumming task at 120 bpm, and moderate dynamic level, for all subjects. Results are presented only for DOF's that showed oscillation (5), and thus for which learning took place. For each DOF, a two second section of the actual (recovered) and learned (reproduced) trajectories are shown (top plot), as well as the residual error of the learning task (bottom plot).

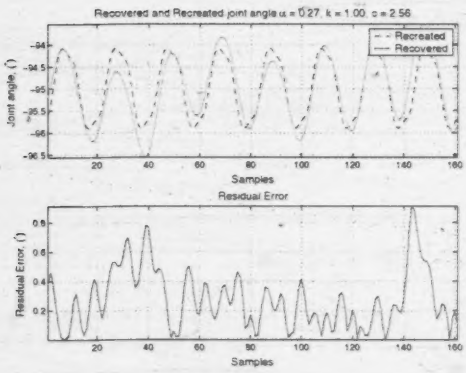
B.1 Subject one results



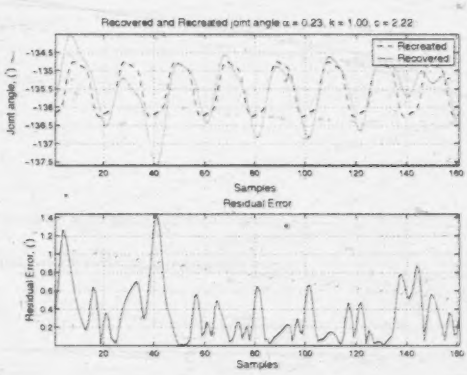
(a) Shoulder X component.



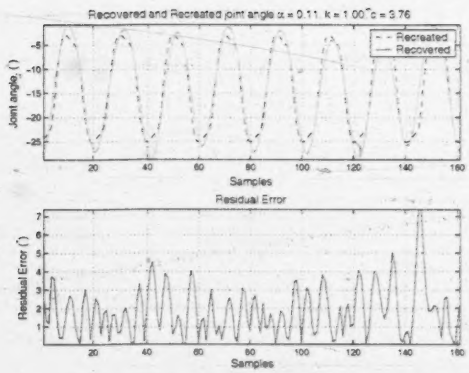
(b) Shoulder Z component.



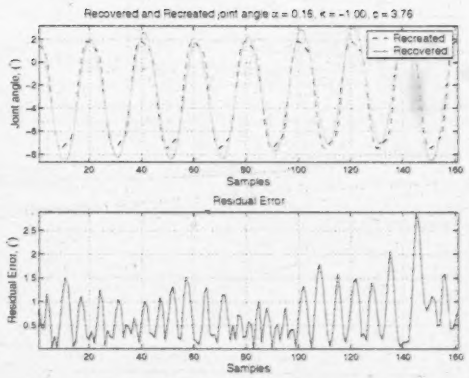
(c) Elbow X component.



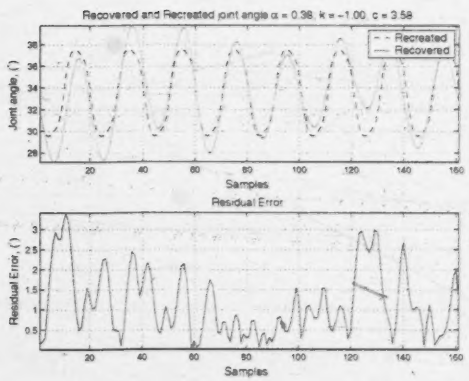
(d) Elbow Z component.



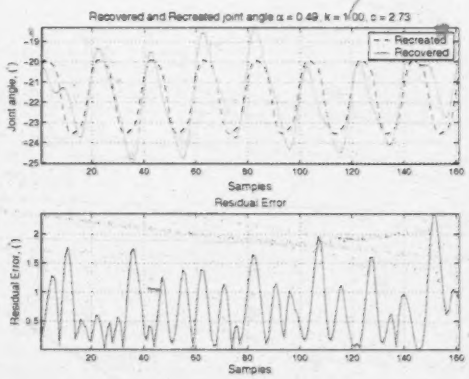
(e) Wrist X component.



(f) Wrist Y component.



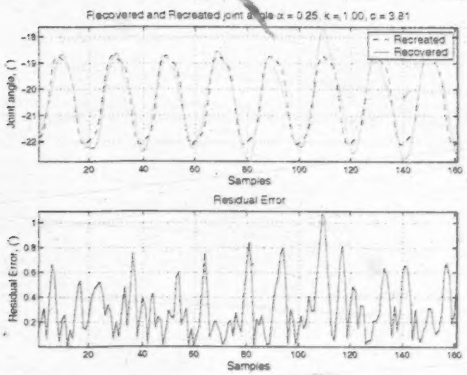
(g) Fulcrum X component.



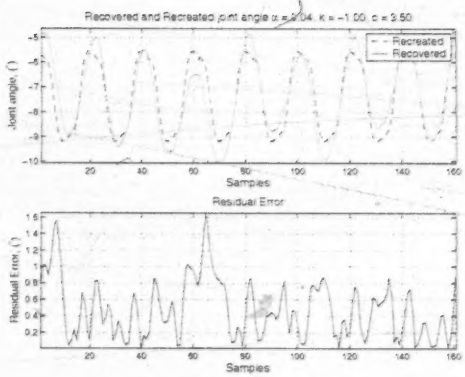
(h) Fulcrum Y component.

Figure B.1: Partial time series of recovered and reproduced joint angle trajectories of single stroke drumming task, subject one.

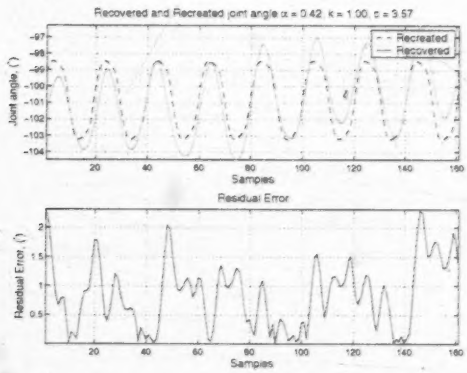
B.2 Subject two results



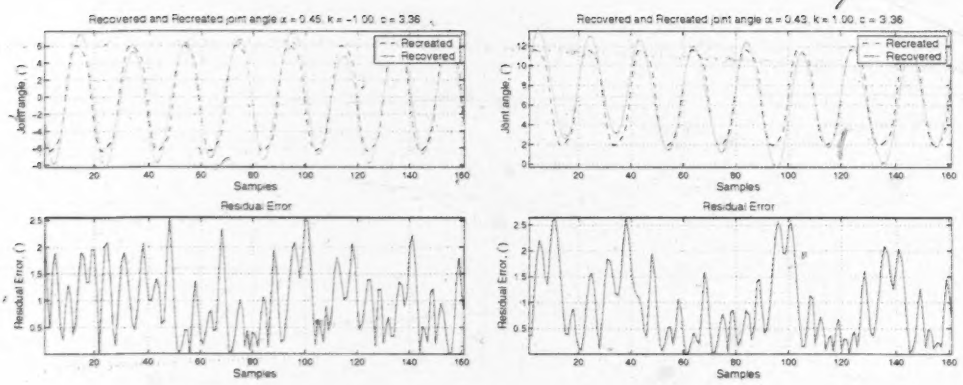
(a) Shoulder X component.



(b) Shoulder Z component.

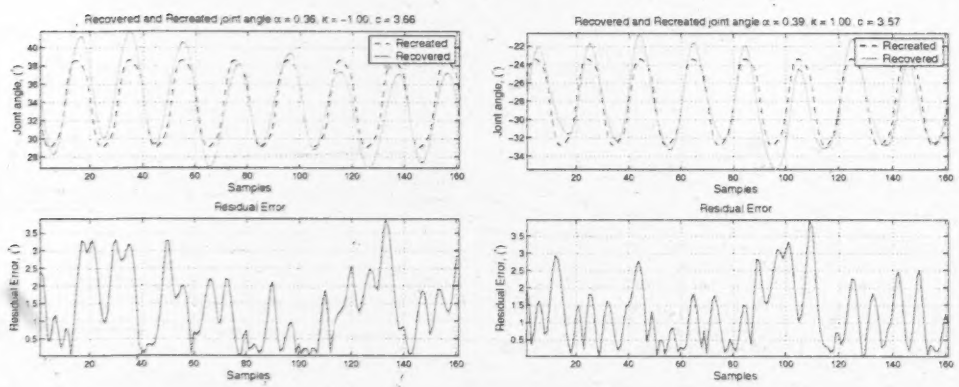


(c) Elbow Z component.



(d) Wrist X component.

(e) Wrist Y component.

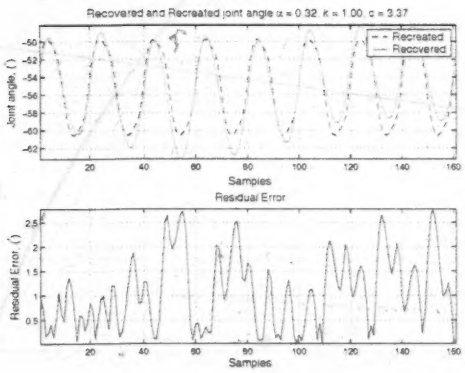


(f) Fulcrum X component.

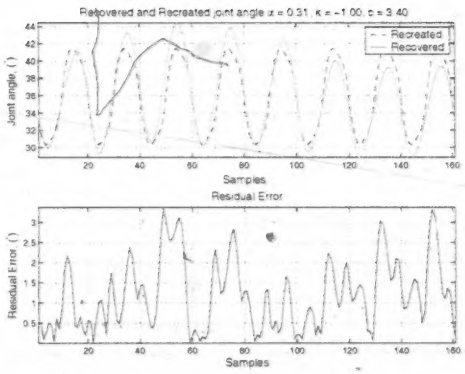
(g) Fulcrum Y component.

Figure B.2: Partial time series of recovered and reproduced joint angle trajectories of single stroke drumming task, subject two.

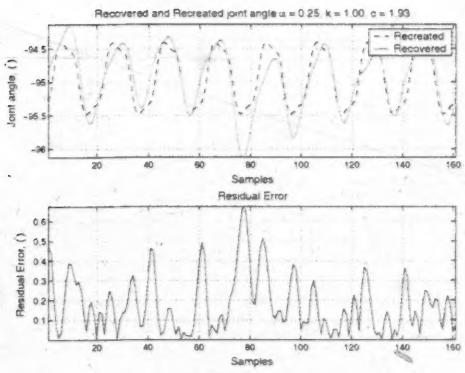
B.3 Subject three results



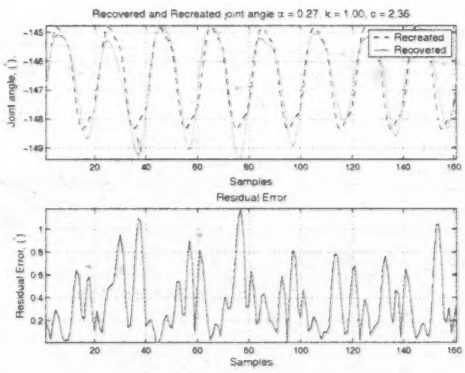
(a) Shoulder X component.



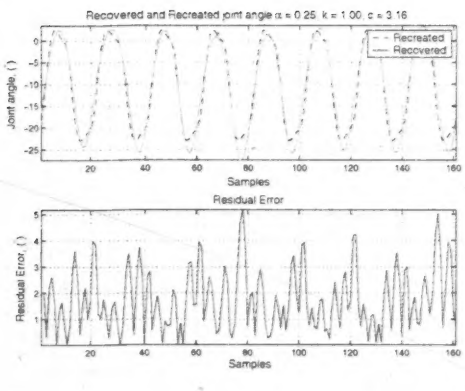
(b) Shoulder Z component.



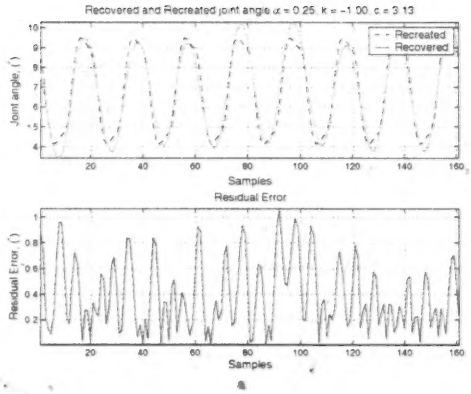
(c) Elbow X component.



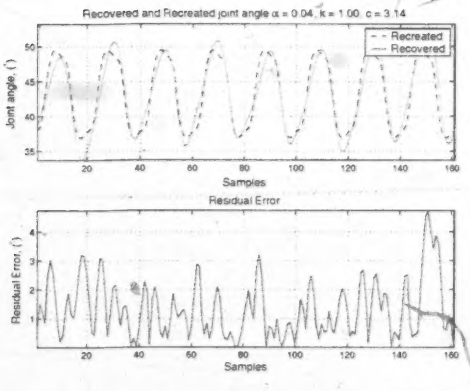
(d) Elbow Z component.



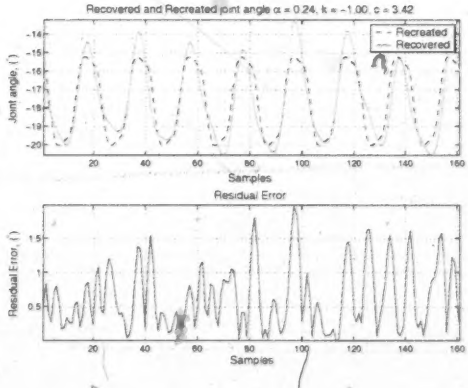
(e) Wrist X component.



(f) Wrist Y component.



(g) Fulcrum X component.



(h) Fulcrum Y component.

Figure B.3: Partial time series of recovered and reproduced joint angle trajectories of single stroke drumming task, subject three.